DEVELOPMENT, VERIFICATION, AND VALIDATION OF THE RESPONSIVE BOUNDARY MODEL FOR POOL FIRE SIMULATIONS

by

Weston M. Eldredge

A dissertation submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Chemical Engineering

The University of Utah

August 2011

The University of Utah Graduate School

STATEMENT OF DISSERTATION APPROVAL

The dissertation of **Weston M. Eldredge**

has been approved by the following supervisory committe members:

Philip J. Smith	, Chair	**May 25, 2011** Date Approved
Milind D. Deo	, Member	**May 4, 2011** Date Approved
Eric G. Eddings	, Member	**May 4, 2011** Date Approved
Aaron Fogelson	, Member	**May 4, 2011** Date Approved
Edward M. Trujillo	, Member	**May 4, 2011** Date Approved

and by **JoAnn S. Lighty** , Chair of the Department of **Chemical Engineering**

and by Charles A. Wight, Dean of The Graduate School.

ABSTRACT

The need to understand and predict the behavior of fires and explosions is important considering the amount of property damage and the loss of life that can result. In the case of transportation pool fires (fires resulting from liquid fuel spills), predictive science is an especially valuable tool considering that experiments with large pools fires are costly and often lead to damaged or destroyed instrumentation. In the development of fire codes as well as in other areas of computational science, the need for fidelity in computational results has become a prominent issue. The various sources of error in computation, such as discretization error, machine round-off error, iterative convergence error, programmer error, and model error, must be accounted for and, if possible, quantified if computational results are to be considered legitimate.

The current study seeks to remedy an important source of error in pool fire simulations. The error stems from the application of a simplistic fuel inlet boundary condition. Traditionally this type of boundary condition assumes that the liquid pool vaporizes fuel to feed the flame at a constant rate. Additionally the vaporization rate is assumed to be uniform over the pool surface. In reality there is a complex feedback mechanism between the pool surface and the flame. Pool vaporization rate changes with time as the pool is heated, and thermal flux to the pool will be nonuniform over the surface. The Responsive Boundary model utilizes energy and mass conservation principles to model the thermal behavior of the fuel pool and to predict the vaporization rate given thermal input from the flame.

Verification tests such as the Method of Manufactured Solutions and grid convergence and validation methods such as model input sensitivity analysis and consistency analysis are applied to the Responsive Boundary model on its own and linked with the gas-phase fire code (ARCHES). The tests verify that the code solves the continuum model that is the basis of the boundary model with acceptable error. A region of consistency is also found between the steady vaporization fluxes predicted from the model and experimental

data for a small heptane pool fire. Consistency analysis is also applied to data obtained from ARCHES simulations of a small helium plume and data taken from holographic interferometric images.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

CFD	Computational Fluid Dynamics
C-SAFE	Center for the Simulation of Accidental Fires and Explosions
DNS	Direct Numerical Simulation
GCI	Grid Convergence Index
HFG	Heat Flux Gauge
LES	Large Eddy Simulation
MMS	Method of Manufactured Solutions
RANS	Reynolds-Averaged Navier Stokes
RI	Refractive Index

NOTATION AND SYMBOLS

A	Area
b_{ii}	Generic quadratic model fit parameters (Equation 5.3)
C_f	Fluid heat capacity (Equation 2.13)
C_i	Concentration of species i in pool liquid
C_p	Heat capacity (Equation 3.1 - 3.3)
$C_{p,g}$	Gas phase heat capacity
C_t	Total concentration in gas/liquid interphase film
C_ε	Consistency measure
d	Pool diameter (Equation 2.3)
$D_{a,b}$	Bindary diffusivity of species a and b
F	Geometric view factor
g, g_i	Gravitational acceleration in the i^{th} direction
h	Heat transfer coefficient
h_m	Mass transfer coefficient
ΔH_C	Heat of combustion
ΔH_{evap}	Heat of vaporization
ΔH_v	Heat of vaporization
ΔH_{vap}	Heat of vaporization
E	Energy
$f^{n+1}(\xi)$	The n+1 derivative of function f evaluated at ξ, used to determine Lagrangian interpolation error (Equation 3.21)
f_{exact}	Exact solution for a continuum equation
f_i	Solution to a continuum model solved on grid i

f_{max}	Maximum sensible heat fraction parameter (Equation 2.21)
$f_{sensible}$	Fraction of thermal input used to heat the pool
F_s	Error enhancement parameter for the GCI (Equation 2.33)
$[GCI]$	Grid convergence index (Equation 2.33)
Gr	Grashof number
Gr_ω	Diffusional Grashof number
h	Heat transfer coefficent from Newton's law of cooling (Equation 2.12)
h_m	Mass transfer coefficient (Equation 2.18)
h	Pool liquid height (Equations 2.22 and 2.23)
H	Pool liquid height (Equation 3.4)
H_{evap}	Heat of vaporization
H_{fg}	Heat of vaporization
$\mathcal{H}$	Hypercube that represents a model input parameter space
$I(z)$	Intensity of electro-magnetic wave as a function of pool depth z
I_0	Incident intensity of electro-magnetic wave as it passes the pool surface
k	Parameter that characterizes temperature profile in the liquid pool (Equation 2.5)
k	Generic constant used in defining Taylor series error terms (Equation 2.26)
k	Thermal conductivity (Equation 3.1)
k_1, k_2	Parameters used to fit pool temperature profile (Equation 3.22)
k_1, k_2, k_3, k_4	Intermediate parameters for Runga-Kutta method (Equations 3.6 and 3.7)
k_c	Mass transfer coefficient (Equation 3.10)
k_f	Fluid thermal conductivity (Equation 2.16)

$[k_i]$	Mass transfer coefficient matrix evaluated at location i
k_l	Thermal conductivity
l	Characteristic length associated with the Reynolds number (Equation 2.2)
l_e	Lower bound of experimental data uncertainty
L	Characteristic length associated with the definitions of the Grashof numbers (Equations 3.11 and 3.12) and the mass transfer coefficient (Equation 3.16)
Le	Lewis number
m	Mass of liquid in pool (Equation 2.24)
$\dot{m}$	Mass vaporization rate of fuel
m''	Mass flux from the pool surface due to evaporation
$m''evap$	Mass flux from the pool surface due to evaporation (Equation 3.4)
n	Correlation parameter (Equation 2.19)
n_0	Standard refractive index (usually for pure air) for use in the optical path length difference relation (Equation 5.1)
n_i	Refractive index of medium i
$n_m(x, y, z, t)$	Refractive index as a function of both space and time, used in the optical path length difference relation (Equation 5.1)
$\mathbf{N}$	Vector of molar fluxes in a multi-component system
$N_{0,\,i}$	Specific refractivity of species i
$N''evap$	Molar flux of fuel from pool surface due to evaporation
$p,\ P$	Fluid pressure
p	Order of grid convergence error (used in Equations 2.30 and 2.31)
$p_k(x)$	The k^{th} Lagrangian polynomial evaluated at spatial coordinate x
P_1	Reference pressure in Clausius-Clapeyron equation (Equation 2.15)

P_{cc}	Clausius-Clapeyron vapor pressure
$P_n(x)$	An interpolated function value as a function of spatial coordinate x, based on a series of Lagrangian polynomials
$P_{s,\,i}$	Saturation pressure of species i for use in VLE model (Equation 3.18)
P_{vap}	Partial pressure of fuel in vapor phase
q	Generic term for heat transfered
q''	Generic term for heat flux
q''_{conv}	Heat flux due to convection
$q''_{convection}$	Heat flux due to convection
q''_{evap}	Energy flux due to evaporation of fuel from the pool (Equation 3.3)
$q''_{incident}$	Incident radiative heat flux to pool surface
q_{mms}	Source term used with manufactured solutions
q''_{re-rad}	Heat flux radiating from the pool surface
$q''_{total,\,pool}$	Net input of heat to the pool (Equations 2.20 - 2.24)
q_ϕ	Source term for the scalar balance (Equation 1.3)
Q_{abs}	Thermal input to the pool surface (Equation 2.13)
Q_c	Heat release due to fuel combustion
Q_f	Thermal input to pool surface (Equation 2.13)
r	Grid refinement ratio (used in Equations 2.30 and 2.31)
r_b	Liquid regression rate (Equation 2.13)
R	Pool surface reflectivity, also universal gas constant
$R_n(x)$	Error associated with use of Lagrangian polynomial interpolation as a function of spatial coordinate x
R_p	Surface reflectivity for light polarized parallel to the interface plane of interest

R_s	Surface reflectivity for light polarized perpendicular to the interface plane of interest
Re	Reynolds number
$S(x, y, t)$	Interference order as a function of spatial coordinates x and y and of time t
Sh	Overall Sherwood number
Sh_{forced}	Sherwood number associated with forced convection
Sh_{free}	Sherwood number associated with buoyancy driven convection
t	Time or temporal coordinate
t_{end}	Time at wihich manufactured solution for the mass balance drops to a value of zero
Δt	Temporal step size
T, $T(z)$	Liquid temperature as a function of liquid depth
T_0, T_∞	Ambient temperature
T_1	Reference temperature in Clausius-Clapeyron relation (Equation 2.15)
ΔT_b	Parameter for pool energy balance model (Equation 2.21)
T_b	Boiling point temperature
T_B	Pool temperature (Equation 2.3)
T_{cc}	Temperature at which Clausius-Clapeyron vapor pressure is computed (Equation 2.15)
T_f	Flame Temperature
$T_{nominal}$	Nominal temperature parameter used in the manufactured solution to the energy balance equation (Equation 4.1)
T_s	Pool surface temperature
T_{pool}	Pool liquid temperature (Equations 2.21 and 2.22)
u_e	Upper bound of experimental data uncertainty
u_i	Fluid velocity in the i^{th} direction

u_∞	Bulk flow fluid velocity
U	Heat transfer coefficient (Equation 2.3)
U	Generic quantity of interest (Equation 2.25)
v	Liquid regression rate (Equation 2.4)
v_∞	Maximum liquid regression rate parameter (Equation 2.4)
V	Volume
$\mathbf{x}$	Vector of input parameters to a model
x, x_i	Spatial coordinate in the i^{th} direction
$\mathbf{x_0}$	Mole fraction vector, liquid phase side of film
x_{bulk}	Fuel mole fraction in the bulk gas phase (Equation 3.10)
x_i	Fuel mole fraction at gas/liquid film (Equation 3.10)
x_i	Liquid phase mole fraction of species i for use in VLE model (Equation 3.18)
$\mathbf{x}_\delta$	Mole fraction vector, gas phase side of film
X_{vap}	Mole fraction of fuel in vapor phase
y	Spatial coordinate
$y(x_1, x_2, x_3)$	Function output for quadratic model as a function of model inputs x_1, x_2, and x_3 (Equation 5.3)
y_i	Gas phase mole fraction of species i for use in VLE model (Equation 3.18)
y_e	Experimental measure of some quantity Y_e
Y_e	Physical quantity of interest to be measured or predicted
$y_m(\mathbf{x})$	model prediction as a function of model inputs $\mathbf{x}$
$Y_{F,\infty}$	Mass fraction of fuel in the bulk gas phase
$Y_{F,s}$	Mass fraction of fuel at gas/liquid interface
z	Spatial coordinate often associated with pool depth, also used along the beam path in holographic interferometry
Δz	Spatial step size in pool domain

α	absorption coefficient defined in Equation 2.8
α	Pool surface absorptivity (Equation 2.20)
α	Thermal diffusivity (Equation 2.27 - 2.29)
α	Parameter used with manufactured solution for the mass balance equation (Equation 4.2)
α_j	Lower bound of uncertainty in model input parameter j
β	Thermal expansion coefficient associated with the Grashof number (Equation 3.11)
$[\beta_i]$	Bootstrap matrix evaluated at location i
β_j	Upper bound uncertainty in model input parameter j
γ	Adjustment parameter for uncertainty bounds in experimental data, used to determine consistency (Equation 2.38)
γ_i	Liquid phase activity coefficient of species i for use in VLE model (Equation 3.18)
γ_i	Mass fraction of species i used with the Gladstone-Dale relation (Equation 5.2)
Γ	Parameter that characterizes a diffusive velocity in the scalar balance (Equation 1.3)
δ	Thickness of gas/liquid film
Δ	Symbol that denotes a change or perturbation in a quantity
∂	Parital differentiation symbol
ϵ	Emissivity
ϵ	Fractional error between continuum model solutions from two different grid resolutions (Equation 2.32)
ε	Experimental data space
θ_i	Angle of incident light
κ	Beer's law extinction coefficient
λ	Wavelength of electro-magnetic wave
λ_f	Fluid thermal conductivity (Equation 2.13)

$\lambda^{(i)}{}_j$	Lagrangian multiplier for j^{th} quantity of uncertainty type i
ν	Fluid kinematic viscosity
ξ	Compositional expansion coefficient associated with the diffusional Grashof number (Equation 3.12)
$[\Xi_i]$	Bulk flow effect matrix evaluated at location i
π	Mathematical constant
ρ, ρ_f	Fluid density
ρ_g	Gas phase density
σ	Stefan-Boltzman constant
τ_{ij}	Value of the stress tensor associated with directions i and j
ϕ	Conserved scalar quantity (Equation 1.3)
χ	Combustion efficiency
ω_i	Mass fraction at location i

ACKNOWLEDGEMENTS

To Doctor Philip Smith for his seemingly unending supply of patience and his willingness to let me figure things out on my own. To my parents Fred and Jannis Eldredge who are largely the reason for what I am today.

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

Early in the morning on April 29, 2007 a gasoline tanker lost control and struck a pylon on a freeway interchange near the San Francisco-Oakland Bay Bridge. According to the Associated Press (2007), the tanker carried 8,600 gallons of gasoline that spilled and ignited. Observers of the accident noted that the resulting flames rose 200 feet in the air. The flames engulfed another highway interchange just above the accident scene. The heat from the flame melted the second interchange and caused a 250 foot section of the highway to collapse to the ground below. The accident destroyed a busy section of highway that serves nearly 100,000 vehicles each day. It required months to rebuild the destroyed section of highway, and in that time, the accident led to massive disruptions of city transportation. Although it was a costly accident, fortunately no lives were lost.

Humankind has made use of fire for roughly 400,000 years. Fire has enabled them to cook food, allowing for an increased intake of nutrition. Fire use has provided a stable and relatively easily manipulated source of heat, allowing for the habitation of colder climates. In more modern times, the use of fire and other combustion reactions has benefited humankind in the generation of power, the advancement of industry, the use of improved systems of transportation, etc. However the use of fire gives rise to issues of safety, and not all fire scenarios are without the loss of life. According to the U.S. Fire Administration (2009), statistics for the year 2008 include the following facts: There were an estimated 1.5 million fires that year. 3,200 civilians and 118 firefighters lost their lives as a result of fire, and 16,705 civilians were injured. Direct property damage was estimated at 15.5 billion dollars. In 2008 fire claimed more human lives than all other natural disasters combined. Most of the lives and damages were results of residential fires. Nonresidential

fires (fires in public buildings, forest fires, and industrial fires) resulted in over 3.8 billion dollars (U.S. currency) in property damage.

To save lives and cost, it is the objective of governmental, educational, and industrial researchers to develop materials and methods for the prevention of fires and the promotion of fire safety. For example, laws have been enacted that govern the layout of building plans to facilitate fire safety. Fire suppression systems have been developed and implemented to respond to fire in a variety of situations. Fire-resistant materials have been developed to protect firefighters in their work. Similar fire-resistant materials are also used in building and facility construction to prevent property damage. Large sums of revenue are invested by governmental agencies to educate the public on fire safety practices. Industrial and military organizations also invest a significant amount of time and money to protect lives and assets from fire damage.

For some branches of research, the objective is to discover the physical laws and mechanisms that govern fires and other combustion reactions in order to aid in the development of methods to promote fire safety and to utilize fire more efficiently for the benefit of society. Controlled fire experiments conducted by researchers seek to explore such issues as the reactions and reaction mechanisms involved in different combustion processes, the important modes of heat transfer involved in the resulting flames, the behavior of the flow fields produced in fires, the different chemical byproducts produced in fire reaction, and so forth. Because of the dangerous nature of fire, it is common for experimental equipment to suffer damage. For example, the measurements of radiative heat flux in a large fire often involve radiometers being engulfed in the flames. The cost of such experiments can be prohibitive, and the larger the fire the more costly the experiment.

The resources of computational science, if sufficiently developed, can provide a more cost effective way of predicting and analyzing the behavior of fire systems. In the past, the complexity of combustion systems has made meaningful simulation of such systems nearly impossible. With the ongoing advancement of computing systems in terms of processing speed, memory storage capacity, and parallelization, these limitations have become more relaxed, and the capabilities of predictive science in fire systems have expanded. The growth of computational science does not eliminate the need for experiment. For computer models

to be of use, they must be developed using experimental data, and computational results must be validated against experiment. Once appropriately developed and proven however, computer models minimize the need for costly experimentation.

1.2 Development of CFD

Modeling combustion systems requires the application of conservative relations for fluid systems. The conservation of mass, momentum, and energy is the foundational model for computational fluid dynamics (CFD). The resulting equations are called the Navier-Stokes equations. These equations can be expressed in differential or integral form. The differential form of the mass and momentum conservation equations are:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \left(\rho u_i\right)}{\partial x_i} = 0, \tag{1.1}$$

and

$$\frac{\partial \left(\rho u_i\right)}{\partial t} + \frac{\partial \left(\rho u_i u_j\right)}{\partial x_j} = \frac{\partial \tau_{ij}}{\partial x_j} - \frac{\partial p}{\partial x_i} + \rho g_i. \tag{1.2}$$

Here, ρ is the fluid density, u_i is the fluid velocity in the i^{th} direction, t is the temporal coordinate, x_i is the spatial coordinate in the i^{th} direction, g_i is the component of gravitational acceleration in the i^{th} direction, p is the fluid pressure, and τ_{ij} is the viscous stress tensor. These two equations are sufficient to describe isothermal, fluid systems with no chemical reactions. Combustion systems, however, are not isothermal and involve thousands of reactions. To describe such systems it is necessary to include scalar balances with the mass and momentum balances. For a combustion system, scalars of interest include, but are not necessarily limited to, energy and species concentration. The differential form of the scalar conservation equation is:

$$\frac{\partial \left(\rho\phi\right)}{\partial t} + \frac{\partial \left(\rho u_j \phi\right)}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\Gamma \frac{\partial \phi}{\partial x_j}\right) + q_\phi. \tag{1.3}$$

In this expression, ϕ is the scalar quantity (energy or species concentration), and Γ is a parameter that describes the diffusive velocity of the scalar quantity. For energy, this would be the thermal diffusivity, while for species balances, this would be molecular diffusivity. The quantity q_ϕ is the sink or source term for the scalar quantity. For species balances this is the term that describes the consumption or generation of a chemical species from a reaction. These equations form the general model needed to describe a combustion system. Whether written in differential form or integral form, numerical techniques have been developed to solve these equations for a variety of systems. The actual computation of pertinent properties from these models is too labor intensive to do by hand, making computation ideal for the solution of these equations for a system of interest.

Even with computers, solving these equations for any general system is not without problems. One prominent issue in the field of CFD is the modeling of turbulent flow regimes. Examination of the momentum balance equation (Equation 1.2) shows primarily two major forces that influence fluid flow: inertial forces and viscous forces. A dimensionless quantity that gages the relative importance of inertial forces to viscous forces is called the Reynolds number, and it can be defined a number ways depending on the physical system of interest. When the inertial forces are small relative to the viscous forces (low Reynolds number flow), the resulting flow field is relatively simple and can be easily computed for most systems. When the inertial forces overwhelm the viscous forces (high Reynolds number flows) the character of the flow field quickly becomes complicated. Ferziger and Peric (2002) described the character of turbulent with the following observations: Turbulent flows are highly unsteady. Flow velocities fluctuate rapidly in all three dimensions. These flows contain a significant amount of vorticity. Turbulence increases the rate at which scalar quantities are mixed. Turbulent flows exhibit a wide range of temporal and spatial length scales.

This last property of turbulent flows is especially troubling for simulation. Turbulent flows contain eddys and flow characteristics that span a wide range of length scales, and the higher the Reynolds number, the larger the range of lengths scales. In such flow systems, in order to capture all the length scales in the simulation, the physical domain must use a sufficiently small mesh resolution. As the Reynolds number increases, the

needed mesh resolution becomes so small that the computational resources become quickly overwhelmed. Even with modern computing capabilities, there are strict limitations in the ability to simulate turbulent flows using the standard Navier-Stokes equations. Using the Navier-Stokes equations to simulate flow fields is called Direct Numerical Simulation (DNS), and while it is the most rigorous among the common models for simulating flows, it is also the most computationally expensive. With current computational resources only turbulent flows with relatively low Reynolds numbers and simple flow geometries can be simulated using DNS.

Instead of solving for the exact flow fields, the flow velocities can be averaged over a certain time period to obtain a time-averaged flow field. Such a method averages out all eddys common to turbulent flows in the flow field and uses special models, called subgrid scale models, to estimate the effects of turbulent eddys on the average flow field. This approach calculates the large scale behavior of the flow, but largely ignores smaller scale flows that occurs in turbulence. This approach is called the Reynolds-Averaged Navier-Stokes Equations (RANS). In terms of detail captured in the flow and the computational expense it is the opposite of DNS. RANS allows solutions of highly turbulent flows, and for many physical situations it captures all the needed physics to accomplish an analysis. For example, if one is interested in capturing the overall flow direction and magnitude, the average force of the fluid on a solid body, the degree of mixing between two fluid streams, or the amount of substance reacted RANS can be useful. RANS, however, has its limitations: the effects of small scale flows and eddys is not perfectly modeled by subgrid scale models, and if more detailed information is needed about a flow, then RANS will not suffice.

Another method for simulating turbulent flow that is a middle ground between RANS and DNS in terms of computational expense and predictive rigor is Large Eddy Simulation (LES). Traditionally, meteorologists have applied LES for weather prediction. Instead of averaging all secondary flows, LES resolves larger scale flows and eddys while averaging smaller scale fluid structures and using subgrid scale models to simulate their effects on the flow field. LES allows simulations to capture some of the details of turbulent flows without overloading computational resource. The development of methods like RANS and LES has greatly expanded the capabilities of CFD to simulate and predict the properties

of turbulent flow systems.

1.3 Pool Fires

A combustion reaction requires mainly three components: (1) a fuel, (2) an oxidizer (usually oxygen), and (3) a sufficiently strong heat source to ignite the reaction. Fuels can be any of a number of chemical species. Most fuels are primarily composed of carbon (hydrogen gas being the most common exception.) Fuels can be solid, liquid, or gas, but combustion is a gas phase reaction. Gas fuels can burn as soon as all other conditions for the reaction are met, but liquids and solids must experience a change to the gas phase first before combustion can occur. Liquids must be vaporized to the gas phase before they can burn. Some solid fuels can sublime directly to the gas phase, but the most common mechanism for solid fuels to convert to gas phase is through chemical decomposition of the solid fuel into gas phase products. This process is called pyrolysis.

There are safety issues to be studied from fires that result from the burning of gases (i.e., natural gas pipeline leak) and solids (i.e., forest fires), but military and industrial groups are especially interested in the effects of liquid fires. As seen the example mentioned at the start of this chapter, great damage to property and life can occur from liquid fuel spills. Transportation accidents and fuel spills at industrial sites are the types of accident scenarios that concern these groups. For example, if an aircraft crashes and spills its fuel and the fuel should ignite, it is of interest to find out what the potential damages and other effects of the fire might be. If a piece of military ordinance were to be engulfed in the resulting flame, it is of interest to determine if that explosive would detonate and how long it would take. It is also of interest to know if such a fire were to occur how close a person could stand to the flame and for how long before the individual experiences burns or if they would burst into flames immediately. Researchers are also interested to know if a fuel spill occurs, how far it would spread. Since most spills occur on grass, dirt, or other porous ground, the absorption of the fuel into the ground would affect how far across the ground the fuel and the fire would spread.

These types of liquid fires where liquid fuel is spilled and ignited are called pool fires, and they are frequently studied. Researchers are interested in studying the way these fires

behave under different conditions. For example, the size of the pool can affect the flow regime (laminar or turbulent) of the flame, the height of the flame, the dominant modes of heat transfer from the flame, and the intensity of the heat output of the flame. The presence of crosswind can also have a significant impact on the behavior of the flame. Researchers often use metal devices equipped with thermocouples which they place in or near the pool fire in order to measure the heat input to the device. These devices are placed in different areas of the pool to measure the heat output from the flame in different region of the pool. These devices are called calorimeters. Another issue of interest in the study of pool fires is the rate at which the fuel burns. Researchers use devices such as thermocouple rakes, pressure transducers, or they can design the pool system to resupply fuel to the pool such that the fuel level remains steady and the burn rate can be inferred from the mass replenishment rate.

Fire experiments have been conducted on pools as small as a few centimeters in diameter to as large as several meters in diameter, and like most fire experiments they can be quite costly. Therefore, the modeling of pool fires with computers using CFD is also widely used. Computational models, when adequately developed, can be a powerful tool in understanding and predicting the behavior of pool fires without the cost of experiment. ARCHES is an example of such a CFD code developed by the research team at the Center for the Simulation of Accidental Fires and Explosions (C-SAFE) at the University of Utah. According to Spinti et al. (2006), ARCHES uses LES and a finite volume scheme to solve the conservative relations for reactive flows. ARCHES is built upon a computational framework code in C++ called Uintah. Uintah enables the use of massive parallelization for computational codes like ARCHES. With this framework in place, ARCHES can run simulations using thousands of processors. ARCHES has been utilized for a variety of combustion-related systems like pool fires, flares, helium plumes, and so forth. It also employs discrete-ordinate methods to calculate the radiative heat flux generated in the fire. This feature is especially important for calculating the heat flux from the flame to an immersed or adjacent object.

1.4 Research Objectives

1.4.1 Objective 1: Boundary Conditions

Computational codes like ARCHES take some variant of the Navier-Stokes equations and discretizes them spatially and temporally. The differential or integral equations become algebraic equations that must be solved for a discretized, three-dimensional domain. A numbers of issues can arise when working with computational domains in codes like ARCHES. The grid must be fine enough to capture the important physics of interest in the physical system as well as the important flow structures, but if the mesh is too fine it can get computationally prohibitive. Also stability issues arise with mesh size. For example, Courant stability conditions require that the temporal step size must not be too large, and the smaller the spatial step size the smaller the time step must be to keep the simulation numerically stable.

An important factor that affects the stability of the computational domain and the accuracy of the predicted data is the use of accurate boundary conditions. Examination of the differential forms of the Navier-Stokes equations (Equations 1.1 and 1.2) show that these are partial differential equations involving spatial derivatives. Any differential equation with spatial derivatives requires boundary conditions for the solution of that equation on a physical domain. For an ordinary differential equation with a spatial derivative, the boundary condition affects the values but not the shape of the function over the domain of interest. For partial differential equations like the Navier-Stokes equations the boundary conditions become even more important, especially in a three-dimensional domain. The boundary conditions are functions in and of themselves that determine property values at the edges of the computational domain. The values of the boundaries can influence the entire character of the solution of the domain in both value and character. In other words, a change in boundary condition can radically change the character of the entire solution. Thus, in order to obtain accurate information about a physical system such as a pool fire it is important to have accurate boundary conditions. CFD codes like ARCHES employ different types of boundary conditions. Some examples include boundaries where fluid velocities, fluid pressures, or temperatures are specified (Dirichlet boundary conditions), and boundaries where the profile of quantities like temperature, velocities, and pressures

are specified (i.e., Neumann boundary conditions.)

An important boundary condition employed in ARCHES and other fire codes that has been a point of weakness in the simulation of liquid pool fires is the inlet boundary condition for the fuel supply to the fire. This type of condition specifies the gas velocity into the domain, and it is not difficult to understand why such a quantity is so important to the accuracy of the fire simulation. The fire depends on its fuel supply in order to burn at all, and the amount of fuel supplied to the flames will influence all its properties. For fires where the fuel is a gas, the boundary is not difficult to specify. One simply needs to know the gas flow rate supplying the flame and program that profile into the inlet boundary condition. However, for liquid and solid fuels the situation is quite different. As already stated, liquid and solid fuels must convert to the gas phase by some process first before they can burn, and obtaining the rate at which the gas fuel feeds the flame requires an understanding of the phase-changing process. The focus of this project is the behavior of pool fires, so only liquid fuels are considered from here on.

In the past, when modeling pool fires, ARCHES would create an inlet boundary at the bottom of the domain to represent the liquid pool. Data from similar experiments are used to obtain an average fuel burn rate for the pool fire, and from that average burn rate a fuel velocity is assigned to each boundary node where the pool is represented. For the entire simulation of the pool fire, that fuel velocity is used for the whole pool domain. In some cases, a ramping effect is applied to the inlet boundary condition so that the fuel velocity starts at a lower value and as the simulation continues the inlet velocity increases to the predetermined value based on experiment. This ramping is done to simulate the effect of the pool vaporizing more slowly at first before the liquid heats up and vaporizing more rapidly as the pool heats to a steady value. This approach to the inlet condition assumes that the pool vaporizes fuel uniformly over the whole area of the pool and assumes a constant average burn rate for the duration of the burn. This approach is easier to implement in a fire code, but the physical reality is quite different from this model.

In reality, there is a more complex feedback relationship between the pool and the fire domain above the pool that affects the rate at which the fuel burns. At the point of ignition, the pool, which contains a mixture of volatile hydrocarbons, is evaporating fuel

at a certain rate and feeding that fuel to the flame. As the flame grows, it transfers heat to the surroundings through the various modes of heat transfer including back to the surface of the pool. The pool absorbs the heat and becomes warmer. As it warms the rate at which the fuel vaporizes increases feeding fuel to the flame to a greater degree. Eventually a steady temperature profile in the pool is reached as the heat input is balanced with the heat loss from vaporization. At that point the mass burn rate becomes steady as well. Another complexity of this situation is that the heat input to the surface of the pool is not constant over time or over the area of the pool, and the surface temperature of the pool is not uniform. Because of this the rate of fuel supplied to the flame will also not be uniform over the surface of the pool.

The objective of this project is to more accurately predict the behavior of this type of inlet boundary condition by developing a mathematical model of the liquid pool. This new model allows for a boundary condition that responds to input from the computational domain the same way a liquid pool responds to the fire above it. This type of boundary condition is called a responsive boundary. As computational resource is always an issue in CFD, the responsive boundary model must not be too expensive in terms of computer resource. The physics present in a liquid pool are complex and assumptions must be made to produce an effective model. At the same time sufficient physics of the physical system must be captured to properly predict the right fuel burn rate without including any unnecessary complexity that would make the model a burden to computer resource. The primary purpose of the responsive boundary is to give an accurate fuel burn rate given the heat input. The estimation of the burn rate includes a mass transfer model for volatile species. Vapor-liquid equilibrium (VLE) is important for this system, and its model requires a VLE model. Because the vaporization rate depends on the temperature of the liquid it is important to model to some degree the temperature profile in the liquid pool. Another complexity to consider is that the pool loses mass as it vaporizes fuel. Thus, the physical domain of the liquid pool will shrink with time. The hottest layers of liquid at the top will vaporize away, affecting the temperature profile. The size of the pool will be an important factor in determining the behavior of the pool's burn rate. The presence of a crosswind over the pool is important and will affect the mode of mass transfer that

governs the vaporization rate of the pool (i.e., buoyancy driven flow or forced convection flow.)

Another complexity to consider is the composition of the fuel. Different fuels will produce different flames with different behaviors. Heavier fuels will produce different burn rates than lighter fuels. Fuels that generate large quantities of soot in the flames with have different heat transfer properties that affect the burn rate than fuel with low soot potential. Several accident scenarios of interest involve the spilling of jet fuel. The common fuel considered is called JP-8. The problem with gasoline, jet fuel, and other common fuels is that they are complex mixtures of hundreds of hydrocarbon fuels and conditioning agents that can vary in composition according to time and location of where they are processed. The best way to treat this complexity is to model the behavior of the complex fuel using a simpler, surrogate mixture. A formulator of a surrogate fuel seeks to mimic certain properties of the complex fuel with a fuel of simpler, known composition. Eddings et al. (2005) and Violi et al. (2002) produce an example of such a formulation. They came up with two surrogates each of which consisted of six components to simulate certain properties of JP-8 such as smoke point, flash point, latent heat of vaporization, and heat of combustion. Modeling multicomponent fuel with hundreds of components whose identity and relative concentration are not fully known is impossible to program into the Responsive Boundary model, but a simpler, six component mixture with known composition is much more feasible.

The goal for the project is to develop the Responsive Boundary model that can capture the important physics of the liquid pool and be able to adequately handle any fuel mixture of interest. The model must be incorporated into ARCHES to be used with pool fire simulations. Once incorporated into the CFD code it will be of interest to see the impact of the new boundary condition on the predictions of the pool fire's properties and behavior.

1.4.2 Objective 2: Verification and Validation

In his brief history of the development of CFD, Roache (1998) notes that for a substantial period of time when computer technology experienced tremendous growth and the various fields of computational physics (including CFD) also grew in like manner, the virtually single-minded focus of computationalists was centered on development of

computational technology to describe more and more complex physical systems, and very little was done to develop the means of quantifying the accuracy of results generated from new computational technologies. For a time, most computational results were accepted if the code compiled without error, the results seemed to make sense, and the data produced gave a good match to experimental data. Roache notes that although some groups gave proper attention to the quality of their code, large numbers of other codes developed suffered from a lack of accuracy in their results during this period. For example, in a study by Hatton (1997), over 100 scientific packages were analyzed over a period of four years. In one set of tests each code was checked line by line for what Hatton calls static faults. In this set of tests, a static fault is an error in the code that would not interfere with compilation or cause a crash upon execution of the code, but it could potentially lead to false results. In a second test, certain codes from one discipline (earth science in this case) were run using the same parameters and the same set of input data to compare the results among the different codes. To be brief, the test results showed the performance of the various codes to be problematic. The first set of tests showed that most codes were inundated with various coding faults, and the second set of tests revealed significant discrepancies among codes using the same computational procedures. What is perhaps most disturbing about these test results is that many of the codes analyzed were proven software. Meaning that they had already been in wide use.

Computational science is a relative newcomer to the field of research along with the experimentalists and theorists. Although computational science has shown great promise for the benefit of research, the example above and other similar lessons in the history of computational science show that it cannot replace experimentation, and if its results are to be deserving of the same regard as those from experiment and theory, they must comply to the same standards of quality. Experimentalists must quantify the error in the results of their work for it to be properly recognized, and there is no reason that the same standard should not apply to the results that come of computation.

Errors in computational models arise from the following sources: The first source is programmer error or coding bugs. As demonstrated by Hatton (1998) coding mistakes are an ever present and abundant source of error to a codes performance, especially in larger

codes. One can spend a significant amount of time and resource finding and eliminating bugs, and yet it seems that there are always more to find. Bugs that prevent compilation are relatively easy to resolve. Bugs that are not detected in compilation and still cause errors are particularly worrisome as their effects may not be apparent. The second source of error in a code is round-off error due to the limitations of finite-precision mathematics common to all computers. Beyond the utilization of stable algorithms, or algorithms that do not enhance the effect of round-off error in the course of their execution, little can be done to eliminate this type of error. As long as stable algorithms are employed in the code, the effect of round-off error is minimal. The third source of error in simulation is the error associated with spatial and temporal discretization of continuum equation models like the Navier-Stokes equations described earlier and the error inflicted from iterative procedures that are not sufficiently converged. The last type of error occurs due to choice of mathematical models that depart in some degree from physical reality due to simplifying assumptions. All but the last source of error represent discrepancies between the mathematical model of the system and the computer implementation of that model. These errors represent a problem with computer programming and algorithm development. The process of quantifying and hopefully minimizing such errors is called verification. The discrepancy between the computer implementation of the model and physical reality caused by the last of the sources of error mentioned above represents problems with the conceptual model. Quantifying this type of error in a code's performance is called validation. The terms verification and validation are given a more formal definition by the American Institute of Aeronautics and Astronautics (AIAA) (1998). They define verification as the process of determining that a model implementation accurately represents the developer's conceptual description of the model and the solution to the model. They likewise define validation as the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model. Trucano and Oberkampf (2002) note that both definitions qualify verification and validation as processes that are ongoing. The processes verification and validation are never complete. One can quantify error in a model for certain applications to a point, but one can never complete these processes for all possible applications. A code's performance is much like a the application

of a scientific theory, it cannot be proven, it can only be disproven. Nevertheless, a persuasive effort must be made to demonstrate the quality of code for relevant applications.

A code must be adequately verified before it can be validated. These processes are necessary for a scientific code to achieve a certain credibility. Roache (1998) notes in his history of verification and validation development that several professional and scientific organizations changed policy in regard to the acceptance of scientific work which included numerical results from a scientific code. For example, in 1986 the ASME Journal of Fluids Engineering made the following declaration: The Journal of Fluids Engineering will not accept for publication any paper reporting the numerical solution of a fluids engineering problem that fails to address the task of systematic truncation error testing and accuracy estimation. Within ten years of this statement several other scholarly journals made similar policy statements. Just as experimental evidence must have error analysis to achieve credibility, so too must computational data.

An objective of this research project is to take sufficient steps in the process of verifying the performance of the Responsive Boundary model and the validation of its results. As noted above, it may not be possible to prove the fidelity of the Responsive Boundary code, but steps can be taken to build reasonable confidence in its performance for the physical situations of interest.

Figure 1.1 illustrates what is called a validation hierarchy. The hierarchy depicted demonstrates the complexities involved in the problem of predicting the total heat flux to an object immersed in a large transportation pool fire. The highest box in the hierarchy represents that very scenario. Moving down the hierarchy the overall problem is broken down into simpler physical phenomena. Each lower box represents a subset of important physical phenomena to be simulated. The higher in the validation hierarchy the more complex the system and the more difficult it is to obtain high fidelity experimental data for model validation purposes. The lower down the hierarchy the physical systems become simpler, and the experimental data available for such systems is higher in fidelity. It is important to validate computational models with the lower hierarchy systems first in order to improve fidelity with the higher systems in the hierarchy.

The current project focuses on a subset of the hierarchy depicted in Figure 1.1. The

validation hierarchy for this project is shown in Figure 1.2. The peak of this smaller hierarchy is the problem of simulating a large liquid pool fire. The problem is broken into two parts. The first is improvement of the inlet boundary condition via development of the Responsive Boundary model. The second box represents validation of the ARCHES code through simulation of laminar helium plumes and collaboration of simulation results with experimental data from such plumes through an experimental technique called holographic interferometry. Helium plumes are excellent systems for validation of pool fires as the two systems have similar flow characteristics.

Chapter 2 of this work establishes the important theoretical and experimental background relating to pool fire research and identifies the important physics that influence the correct prediction of liquid pool burning rate, and it discusses previous attempts at liquid pool models. Chapter 2 also discusses common methods of verification, validation, and uncertainty quantification. Chapter 3 describes the formulation and application of the Responsive Boundary model. Chapter 4 describes verification and validation activities related to the Responsive Boundary model prior to its application with the ARCHES fire code. Chapter 5 describes the technique of holographic interferometry and the efforts to validate ARCHES calculations with holographic data from helium plume experiments. Chapter 6 describes verification and validation activities of the ARCHES code with the Responsive Boundary model incorporated.

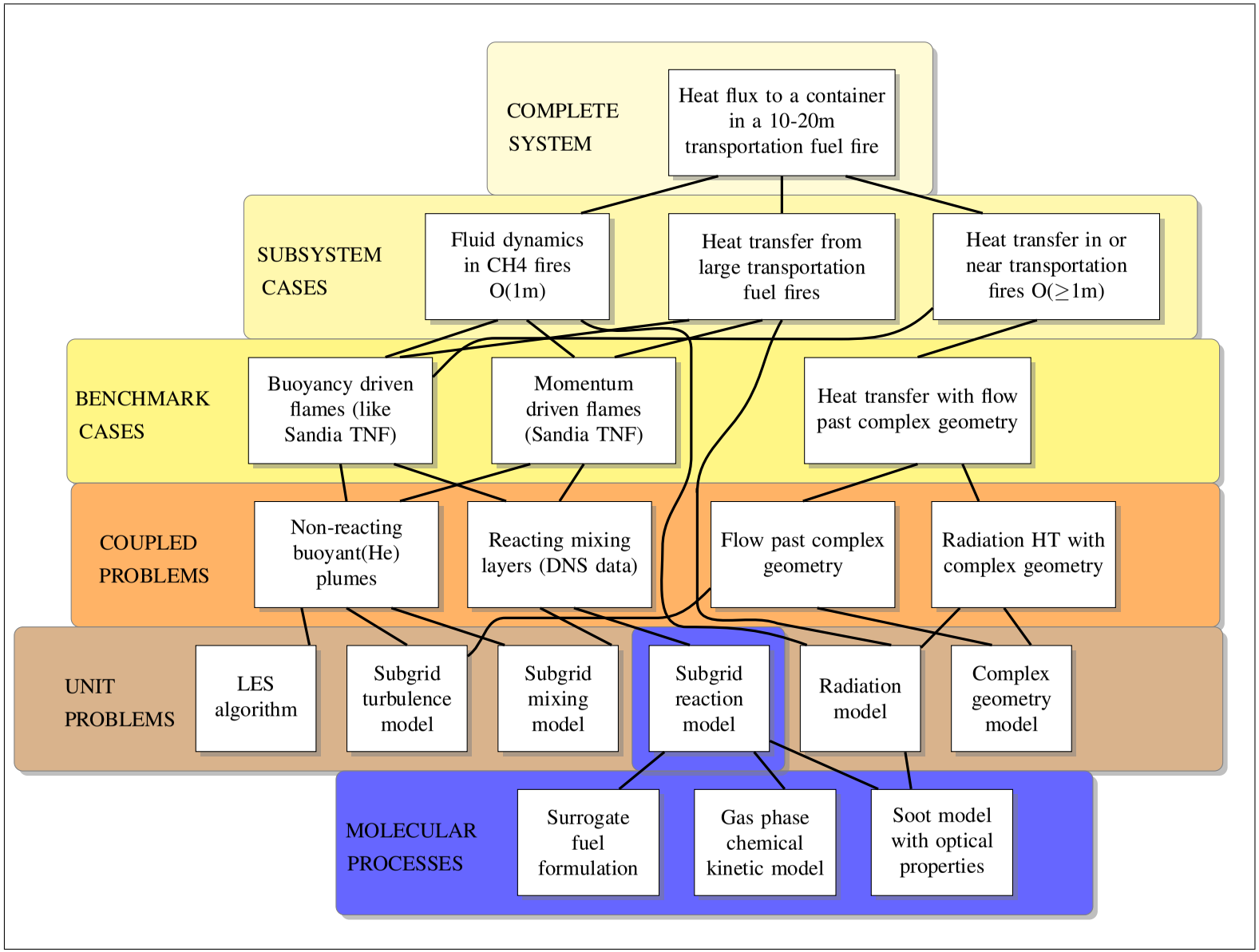

Figure 1.1: The validation hierarchy for the problem of determining heat flux to a container engulfed in a large transportation pool fire.

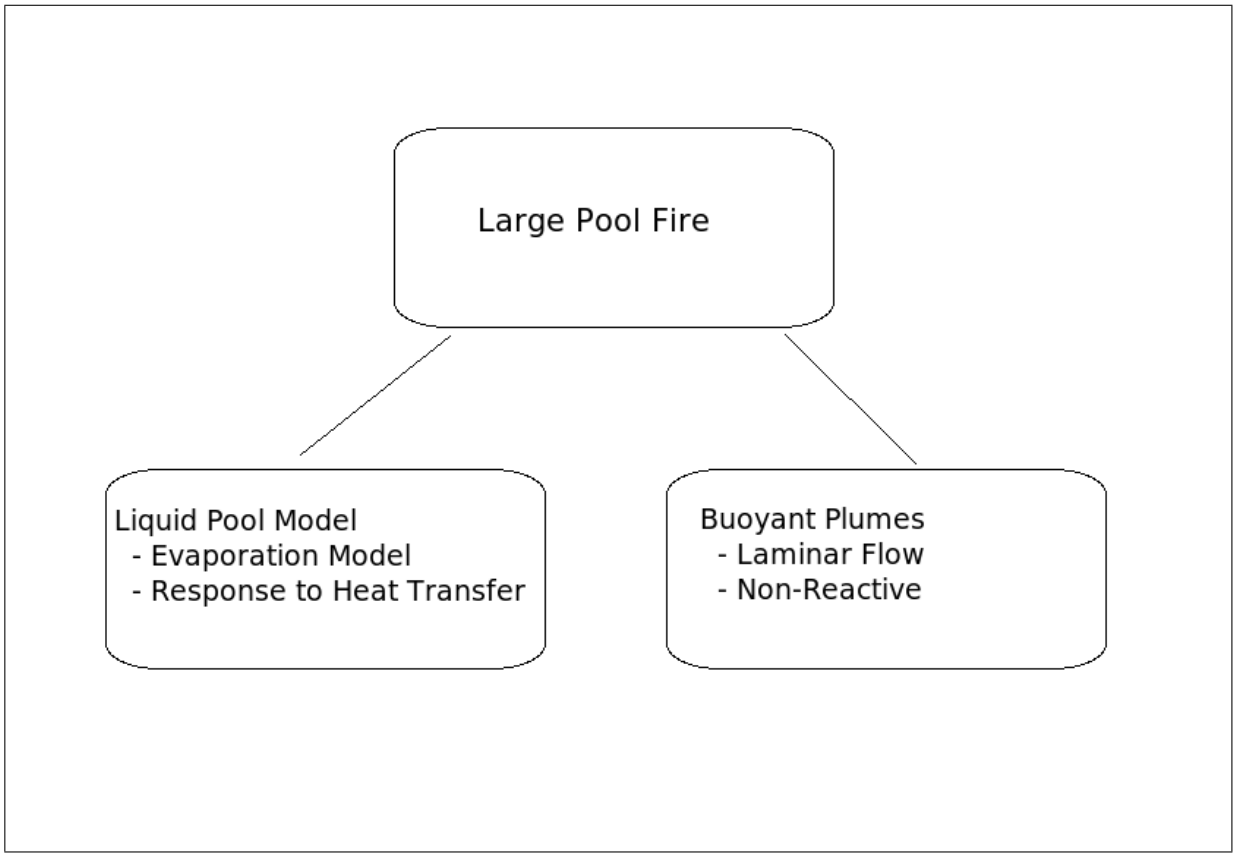

Figure 1.2: A smaller hierarchy where the efforts of the present research are focused.

CHAPTER 2

THEORETICAL BACKGROUND

2.1 Introduction

In order to construct an effective model of a burning liquid pool it is essential to understand the physical system of the pool fire including the physical phenomena that influence the burn rate of the pool. With this in mind, a review of past research conducted with pool fires is appropriate. The primary objective of the Responsive Boundary model is to give accurate fuel evaporation predictions as boundary condition input to fire code simulations. As the literature is reviewed, the physical phenomena that most strongly affect the pool evaporation rate must be identified and distinguished from the physics that are less important to the this research objective.

Several pool fire experiments in the literature seek to measure the evaporation rate of the liquid pool and determine the important physical factors that influence the burn rate. The researchers reasons for studying the evaporation rate is best understood in the light of safety concerns. In the scenario of an accidental fire, it is critical to know for the given conditions of the fire how the fire will spread, and it is important to be able to predict how much heat the fire will contribute to the surroundings and where the fire will distribute its thermal output. The burn rate of the pool is a critical factor in predicting these phenomena. According to Blanchat et al. (2006) the fuel vaporization rate is among the most important variable to be predicted in fire simulation as it determines how long the fire burns. Drysdale (1998) points out that in addition to the fire duration, the fuel vaporization rate is critical in predicting the size of the fire and the amount of heat released to the surroundings as he demonstrates in the following relation:

$$Q_C = \dot{m}\chi\Delta H_C. \tag{2.1}$$

In this relation, Q_C is the heat release due to combustion of fuel, $\dot{m}$ is the vaporization rate of the fuel, χ is an efficiency that represents the effect of incomplete combustion, and ΔH_C is the heat of combustion for the burning fuel.

Almost all the research on pool fires that measure the liquid vaporization rate focuses only on modeling the global burning rate of the pool. Several sources in the literature acknowledge that the vaporization rate varies spatially over the pool surface as well as with time, but as they are primarily interested in the overall heat output of the fire, the spatial variations are not as important to their purposes. While the prediction of heat output from a given accidental fire is the long-term goal, the immediate goal of this project is to improve the accuracy of fire code simulation by providing a more accurate inlet boundary condition, and this requires predicting the vaporization rate with the spatial and temporal variations intact. Such an improved boundary condition can conceivably increase the accuracy in the predictions of the behavior of the fire including the flame size and shape as well as the directional properties and intensities of the flame's heat fluxes.

2.2 Review of Pool Fire Research

The history of pool fire research goes back at least as far as the 1950s where the first important mechanisms governing pool fires are identified, and it extends to the current time where large, costly experiments are performed with more modern, sophisticated instrumentation in order to provide more detailed and accurate measurements for the purposes of simulation validation. The following review of the research concerning pool fires is by no means exhaustive, but it does include a good, representative outline of the important experiments and findings that have occurred over the past 60 years. In addition to the literature cited here, other good reviews of the progress of pool fire research can be found from Koseki (1999) and Joulain (1996 and 1998).

In order for the literature to be viewed in the right context it is helpful to discuss some of the common experimental conditions for pool fires. One of the early realizations about pool fires burn rates is its dependence on the size of the pool usually characterized by the pool's diameter. Different mechanisms of heat transfer affect the burn rate, and the dominant mode of heat transfer is determined by the size of the pool. For the experiments mentioned

in the literature the size of the pool could be as small as a fraction of a centimeter to as large as several meters ($\sim$ 30 meters) in diameter. The pools themselves are frequently contained in a vessel, pan, or other type of container. Some of the vessels are made from ceramic materials, others use stainless steel. For smaller pool fires, cooling water is run through the lower portion of the pan to control the size of the flames in the fires experiment. For larger pool experiments the same purpose is served by floating a relatively thin layer of fuel over a layer of water. In some of the experiments the fuel level in the pan was kept constant by feeding additional fuel to the pan from a reserve supply to replenish the evaporated fuel. Such experiments are called steady pool fire experiments. With other pool fires, experimentalists simply fill the vessel with a specified amount of fuel and allow the fuel to burn without replenishment until the fuel is exhausted. These types of experiments are referred to as transient pool fire experiments. Another important phenomenon that affects burn rate is the presence of crosswind. In some experiments it is desired to have minimal wind effects and great effort is made to ensure that no external air fluctuations occur during the experiment. In other cases the wind is produced artificially and controlled to give a desired wind velocity and direction. Yet in other cases, especially with larger pool fires, the experiments are conducted in an open, outdoor environment. In these cases the wind speed and direction are carefully measured to account for the winds effect on the pool fire.

The first major collection of research on the nature of liquid pool fires was conducted by two Russian researchers: V.I. Blinov and G.N. Khudyakov (1957). Their work was later reviewed by Hottel (1958) before their report was translated to English in 1961. Hottel calls attention to the importance of their work and derives an important relationship, based on their data and analysis, between the energy balance of the pool and the vaporization rate of the pool. The report by Blinov and Khudiakov details the results of experiments performed by themselves and others, and they identify important pool fire mechanisms and derive mathematical models to predict various behaviors of pool fires. They conduct numerous pool fire experiments ranging in size from less than a centimeter to nearly 23 meters in pan diameter. Their experiments include the burning of common industrial fuels such as gasoline, tractor kerosene, diesel oil, solar oil, and transformer oil. The author's report

includes findings on the size and shape of pool fire flames, the rate of pool vaporization versus the rate of combustion reaction in the flame, the behavior of flame pulsations and its relationship to the Reynold's number for pool fires, the different flow regimes (i.e., laminar, transition, and turbulent) of pool fires, temperature profiles and radiation properties of the flames, the phenomenon of soot formation, compositional changes in fuel mixtures during the burning process, the formation of what the authors call the homothermal layer and the phenomenon of boil over, the transient behavior of the fuel vaporization rate and its relationship to the pool diameter, the effect of fuel composition, oxygen concentration, wind speed, and fuel level with respect to the burn lip on the fuel vaporization rate, the variation of vaporization rate over the surface of the pool, the variation liquid temperature over the pool surface, and the vertical variation of temperature in a burning liquid. Frequently quoted by subsequent researchers, the work by Blinov and Khudiakov is widely considered to be foundational to the area of pool fire research.

Contemporary to Blinov and Khudiakov is the work of Rasbash et al. (1956). The authors in this early and important work perform pool fire experiments with an alcohol mixture, benzole (an aromatic mixture), and petrol (gasoline) in pans that are 30 centimeters in diameter. The authors use thermocouples to measure the liquid temperature at and below the surface during the burning process. They observe the size and shape of the different flames, and note the difference in flame shape between the alcohol and other hydrocarbon fires. The authors are the first to describe the formation of the fuel vapor zone above the pool surface and how it changes during the different phases of burning and between the different types of fuel. They note that especially for the nonalcohol flames the dominance of radiation heat transfer to the pool surface over convection and conduction for 30-centimeter fires. They also give estimates of the radiation flux to the surface based on mathematical models and measurements of flame size, temperature, and emissivity.

Burgess et al. (1961) perform a series of transient pool fire experiments. The fires range from 7 centimeters in diameter to 2.4 meters in diameter, and include fuels such as methanol, hexane, xylene, and what the authors call benzene (its likely that this fuel was actually benzole, an aromatic mixture.) The authors objective in this work is to corroborate the finding of Blinov and Khudyakov (1958) with regard to the dependence of

fuel burn rate on diameter and wind speed. The authors observe the transient warm up period and the steady state period of the pool burning process. They find that the steady burn rate reaches an asymptotic value once the diameter is sufficiently large, and they find that the wind speed increases the steady burn rate of the pool.

Akita and Yumoto (1965) conduct pool fire experiments with methanol in concentric vessels ranging in size from 10 centimeters to 30 centimeters in diameter and in regular pans ranging from 1 centimeter to 60 centimeters. Unlike the experiments that Burgess et al. (1961) describe, these are steady experiments where fuel is supplied to the pan to keep the liquid level steady during the burning process. The authors find from the concentric ring pools that for the smaller fires ($\leq$ 30 centimeters), the fuel burns fastest at the edge of the pool and the burn rate decreases as you move to the center of the pool. They also find that this pattern of burn rate is less prominent in larger pools. They also find that thermal radiation to the pool surface for small methanol fires can be neglected for the pools energy balance. They also confirm the validity for small pool fires of the mathematical model described by Hottel (1958) for predicting global pool burn rate.

Thomas et al. (1965) perform experiments on larger (91-centimeter diameter) ethyl alcohol pool files. In this study, the authors are primarily interested in issues of air flow, air entrainment, and oxygen concentration in the flame region. However, they also measure heat flux to the pool surface for these larger fires. They estimate that at the center of the pool one fifth of the thermal heat flux is due to convection and the rest is radiation while at the edge of the pool convection and radiation share equally in the thermal input to the surface.

Corlett and Fu (1966) study the effects of variation in liquid temperature, vessel wall conductance (thermal conductivity multiplied by wall thickness), and water contamination of the pool on the burn rate. The experiment includes pans ranging in size from 0.6 centimeters to 30 centimeters, and involves the burning of ethanol, methanol, and acetone. They find the effects of these three phenomena on mass burn rate to be negligible. What is perhaps more relevant to the present research is that they make 1 of the first attempts to measure radiation to the pool surface directly using a small insulated vessel containing fuel. The authors also note that radiation flux to the surface becomes important at 5 centimeters

to 10 centimeters diameter. They note that convection heat transfer dominates at the edge of the pool while radiation dominates in the center of the pool which is in agreement with other studies.

Alger et al. (1979) make some very interesting observations concerning large (3.05 meter diameter), transient pool fire experiments that they conduct. They burn methanol and JP-5 (jet fuel) in separate experiments to contrast the behavior of luminous (JP-5) pool fire flames to that of nonluminous flames (methanol). Besides flame structure, the authors also seek to measure radiative feedback to the pool surface. The authors find that nonluminous flames produced higher flame temperature than that of luminous flames due to lower heat losses. Their measurements of radiation feedback indicated that for methanol the radiative input to the center of the pool surface was 75 to 80 kw/m^2 while the JP-5 flame generated 100 kw/m^2 in the same location. They estimate the convective flux to the surface to be about 8.3 kwm^2 on average thus confirming the dominance of radiative feedback for large pool fires. What is also of great interest is that the authors comment on the optical properties of fuel pools. They find that the pool absorbs the thermal radiation effectively enough to only allow negligible penetration into the pool's depth. They estimate the penetration depth to be on the order of 1 centimeter. With regard to the reflectivity the radiative heat flux on the pool surface, the authors comment that most fuels have a refractive index between 1.4 and 2.0 leading to a rough estimation of pool reflectivity between 0.05 and 0.10.

Shinotake et al. (1985) conduct transient experiments of moderate to large sized heptane pool fires (30 centimeters to 100 centimeters in diameter.) The authors measure radiation flux to the pool surface using Gardon type flux meters. In this study, the authors examine the transient behavior of radiation flux as the pool fire progresses. They note an overshoot in the radiative flux during the warm-up period of the fire experiment which they attribute to the formation of the fuel vapor zone that eventually blocks a portion of the radiation to the pool surface once established. According to the authors, this also explains why the feedback radiation tends to level off with increasing diameter while the radiation to the surroundings continues to rise with pool diameter.

Klassen and Gore (1992) and Hamins et al. (1994) conduct a series of important pool

fire experiments with detailed description of the important phenomena (structure of flames, radiation feedback to the pool surface, burning rates, etc.) Their pool fires range in size from 7.1 centimeters to 1 meter using both simple and concentric ring burners. They burn a number of fuels, but much of their analysis focuses on methanol, heptane, and toluene as representative of nonsooting, moderately-sooting, and heavily-sooting flames, respectively. The authors measure flame properties such as flame temperature and soot formation for the soot forming flames. They also measure the optical properties of the flames (i.e., absorption and transmission of thermal radiation.) The authors are also interested in measuring the effect of turbulent motion in the flames on soot formation and temperature. Using narrow view radiative heat flux gauges they measure radiation input to the pool surface, and are able to measure angular dependence of the radiative input. They note for 30-centimeter fires that radiation flux is strongest at the center and declines toward the edge of the pool. The decline is more prominent for the methanol flames. They also note that radiation dominates the heat input for smaller pool diameters for toluene and heptane fires more than for those of methanol fires. They also note the presence of soot particles contaminating the pool surface during the burning process which could conceivably alter the optical properties of the pool. The authors also comment on the importance of the reflection of radiation in considering the pool's thermal balance. They also comment on the importance of re-radiation or thermal radiation originating from the pool surface. The data from this experiment is used to calibrate and validate models similar to the one being developed for the present research objective.

Chatris et al. (2001) burn gasoline and diesel oil in pans of one, three, and four meters diameter. Their fire experiments are transient. Their objective is to measure the fuel burn rate as a function of pool diameter and wind velocity. They observe three distinct phases of transient burning: the transient warm-up phase, the steady burning phase, and the transient burn out phase. They report that the warm-up phase takes about 40 to 60 seconds for the fuels tested. They note that the phenomena of boil over is observed for gasoline and especially for diesel oil. They also note that the effect of wind on the burn rate is not significant until the wind speed reaches two $\frac{m}{sec}$.

Randsalu et al. (2004) conduct numerous JP-8 (another jet fuel) fires in 2-meter

diameter pans under conditions of 13 $\frac{m}{sec}$ wind. Their purpose is to measure fuel burn rate using a number of different experimental techniques. Their analysis focuses on the merits and disadvantages of each technique.

Blanchat et al. (2006) perform four large-scale ($\sim$ 8 meters in diameter), transient, JP-8, pool fire experiments in outdoor conditions at different wind conditions. Their objective is to obtain a variety of measurements for the purpose of aiding in the validation of computational pool fire models. Their measurements include flame and pool temperatures, radiation flux to the pool and to the surroundings, and to immersed and nearby calorimeters. They also closely measure the wind conditions. Flame size and shape are monitored using cameras. Mass burn rate is also measured using a thermocouple rake.

Brown et al. (2006) conduct a phenomenology and identification and ranking exercise (PIRT) to determine the important physics in pool fires that affect the fuel burn rate in preparation for the development of a liquid pool model similar to the one being developed for the current research project. The factors they considered in their analysis include radiation properties of the pool and of the flame region, the effects of fouling of the pool surface, the effects of fuel volatility and vapor pressure, convection within the pool, the effects of wind over the pool surface, the effects of the pools temperature profile, the thermodynamic properties of the fuel, the composition of the fuel, and the presence of materials near the fuel. The authors classify each phenomenon according to its importance in the determination of burn rate. It is also important to note that for many of the phenomena their effect on burn rate was not well understood.

It is interesting to note how the objectives and the methods of pool fire research evolved over time. For example, in the earlier experiments the goal was to identify and gain a general understanding of the important mechanisms that control liquid fires. The researchers had relatively crude instrumentation and they could not take detailed measurement of physical properties in conditions of turbulence common to the larger pool fires. The earlier researchers could not directly measure radiation flux or soot concentrations. Many of the early observations made applied to the global behavior of the fire. They could describe the overall burn rate or the general shape, size, and appearance of the flame, but they could not immediately measure these same flame and pool properties

locally with the same certainty as with the global measurements. As time progressed, the instrumentation became more sophisticated and more detailed information about the physical properties could be measured. The objective of pool experimentation was no longer just about understanding the physics of fire. With developments in computational technology, computational prediction became another valuable tool in understanding fire behavior, and experimentalists then sought to measure more specific data in order to help develop, calibrate, and validate computer models.

2.3 Important Pool Fire Measurements

As radiation flux has a very important influence on the rate of pool vaporization, especially in larger pool fires, it is worth reviewing some of the experimental methods employed to measure this quantity. Some of the earlier researchers such as Rasbash et al. (1956) infer the value of the radiation flux to the pool surface by measuring the the fuel depletion rate and employing simple global radiation models that depend on the flame's temperature and emissivity. Given the different sooting tendencies of the different fuels and the difficulty of measuring flame temperature accurately in turbulent flow conditions as well as the errors stemming from assumptions in the emissivity models such methods of estimating radiation flux are highly prone to error. The next technique is employed by Corlett and Fu (1966). In their experiment, they attempt to measure radiation flux to the pool surface by placing a small, insulated well in the pool at different locations. The well contains a portion of the fuel. The fire is ignited for a time and then extinguished. The level of fuel inside is measured before and after the burning period to determine how much fuel was burned. The radiation flux is calculated from an energy balance on the fuel in the well. Since the well is insulated, heat loss by conduction is considered negligible. One problem with this method is that both convective and radiative heat transfer cause the change in liquid level, and correlation must be used to estimate the convection heat transfer which brings substantial error to the calculation. This particular method can give measurement of radiation at different locations of the pool surface assuming the pool itself is sufficiently large to prevent flame disturbance from the well's presence, but it cannot measure instantaneous flux. Instead, it can only give a time average flux to the measured

point on the surface.

In the studies conducted by Alger et al. (1979), Shinotake et al. (1985), and Hamins et al. (1994) the measurements of radiation flux to the pool surface are taken using devices that use Gardon-type sensing elements. The sensor in these devices is a thin foil which is connected to a sensor body at the edges and a thin wire at the center. The two connection points act as the hot and cold end of a thermocouple system. When heat flux is incident on the center of the foil a signal is generated that is proportional to the heat flux incident on the sensor. The Gardon-type sensor is usually water cooled to prevent heat conduction input to the sensor, and the device is built to withstand the high temperature environment of the flames. One significant problem with the sensor is that the incident heat flux includes both radiative and convective heat flux. A method must be found to separate the two types of flux in order to properly measure the radiative flux. In their experiments Alger et al. (1979) use a heat flux device with a sapphire window which allows radiant flux to pass but prevents the hot gases from entering the chamber and interfering with the sensor. The issue with this design is that the window itself could heat up and contribute its own radiative heat flux to the sensor. A different approach using the same sensor is employed by Shinotake et al. (1985). Their approach utilizes two sensors with the same construction except that one sensor surface is made black in order to absorb virtually all incident radiation, and the other sensor's surface is polished to give it a lower emissivity. As long as certain properties of each sensor are known it is possible to solve a small system of equations that yield the convective and radiative heat fluxes. Two problems with this approach are that the uncertainty in the measured fluxes are limited by the uncertainty in the measurements of the properties of the sensors, and the convective heat flux to the sensor's surface will be different from the convective flux to the surface of the pool as each surface has different properties that affect convection. Hamins et al. (1994) use a device with a small opening for radiative flux to enter. They also purge the opening by blowing nitrogen gas out the opening. The small opening allows the measurement of heat flux at different incident angles depending on the orientation of the placement of the gauge at the pool surface. The nitrogen purge prevents gas product from entering the gauge chamber and depositing material on the sensor. It also reduces convective flux to the sensor. In

their large pool fire experiments Blanchat et al. (2006) make use of a different instrument called a Heat Flux Gauge (HFG). The HFG is a well insulated, cylindrical device that uses a very thin metal plate with a thermocouple attached. The plate must be sufficiently thin such that there are only negligible thermal gradients in the plate. As the plate is exposed to the environment of the fire the temperature of the plate is measured. From the time data of the plate temperature the radiative heat flux is calculated from a heat balance model of the plate. The HFG was designed to measure steady-state radiative heat flux in a fire environment. An uncertainty analysis of the HFG by Blanchat et al. (2000) showed that in validation tests of the device uncertainty in radiative flux measurments could be as high as about 40 kw/m^2. The authors identified the largest source of the uncertainty as unaccounted physics in the gauge's thermal model. The measurement of radiation flux is a difficult and ongoing processes. Examination of heat flux data from a turbulent fire reveals data that vary erratically with time. Any measurement of the flux will undoubtedly have large uncertainty.

Fuel vaporization rate is another important quantity in pool fire experiments whose measurement is worth examining given that the focus of the present study to accurately predict this very quantity. It is worth noting that there are number of ways that the fuel burn rate is quantified. One way is to measure the liquid regression rate which is defined as the fall in liquid level with time and it employs units of length per time. The other way is to measure the change in fuel mass with time, and this is generally as the mass flux with units of mass per unit area per time. There are four techniques mentioned in the literature for determining how fast the fuel is burning. The first technique is visual inspection of the liquid level as it drops with time. This technique can also employ cameras to record the liquid level with time. Since the pool level is likely to fluctuate rapidly during the burning process it is more practical to connect a small glass meter to the pool whose level will be far more stable. This technique is best suited to transient pool fire experiments. For steady experiments where fuel replenishment keeps the pool level constant, the level in the reserve tank can also be monitored using this method. The next technique is to measure the mass of the pool or reserve fuel tank with time using load cells. Data of mass measurements with time can be used to calculate the mass burn rate from the slope of the data. Another

technique is to measure the liquid height with time by measuring the liquid pressure with pressure transducers place at the bottom of the pool. The last technique is an approach that has been developed more recently and used by Randsalu et al. (2004) and Blanchet et al. (2006). The approach uses an array of thermocouples commonly referred to as a thermocouple rake to measure the temperature of the fuel throughout its depth and the gas zone above the pool surface as the depth of the pool changes. Obviously, this approach is best suited to transient pool experiments. Since the temperature measured by the array rises sharply once above the pool surface it is simple to see the pool's surface position as a function of time from the temperature data. Randsalu et al. (2004) performed regression rate measurements using all four described techniques. They found the results for all the techniques to be comparable with the exception of the pressure transducer method.

2.4 The Pool Energy Balance

In their classic work, Blinov and Khudyakov (1957) examine the evolution of the fuel in a pool fire. The fuel starts in the liquid phase, and as the fire begins to warm the pool the fuel then vaporizes into the gas phase. Once the fuel rises from the pool surface it mixes with the air, and once the fuel/air mixture is in the proper ratio the fuel reacts generating the proper chemical products. Blinov and Khudyakov are interested in determining how fast the fuel burns. They compare the rates of the different processes in the fuel's evolution and find that the vaporization rate is slow compared to the reaction rate of the combustion process. This means that the burning rate of the fuel can only go as fast as the pool supplies fuel to the flame zone through the process of fuel evaporation. The rate of combustion is also limited by the supply of oxygen to the flame. Blinov and Khudyahkov find that the rate of pool vaporization depends on the heat input to the pool surface. In his review of their data Hottel (1958) identifies three sources of heat transfer to the pool from the flame: Radiation from the flame, convection to the pool surface due to high temperature gases adjacent to the pool, and conduction to the pool through the heated experimental vessel in which the fuel is contained. The pool loses energy through the process of evaporation. Figure 2.1 shows a rough schematic of an experimental pool and demonstrates the three sources of heat input to the pool.

Also, in their experiments Blinov and Khudyakov measure the steady, liquid pool regression rate as function of the pool's diameter. They find that the steady regression rate is highest at small pool diameters and drops sharply with increasing diameter. The regression rate then flattens out to a minimum value (around 10 centimeters diameter for most fuels) and begins to rise again. The regression rate rises and then reaches an asymptotic value at a certain pool diameter that is generally above 1 meter diameter for most of the fuels tested. Figure 2.2 shows a qualitative graph of the regression rate with pool diameter based on the work of Hottel (1958).

Note that in Figure 2.2, the curve varies in the critical lengths from fuel to fuel. Also note that the flow regime regions are labeled with the regression trend. As mentioned before the Reynold's number is what generally characterizes the flow regime of a given flow system. Its mathematical definition for a pool fire is:

$$Re = \frac{u_\infty l}{\nu}. \tag{2.2}$$

Here, Re is the symbol for the Reynold's number. u_∞ is the system velocity. For the pool fires the velocity is the fuel velocity as it leaves the pool. The symbol l represents some characteristic length that depends on the geometry of the system. For the pool fire system, the characteristic length is the pool's diameter. Lastly, ν represents the fluid kinematic viscosity. Although the Reynold's number traditionally represents the flow regime in the case of pool fires, Blinov and Khudyakov choose to represent the flow regime by the pool diameter alone.

In their explanation of the regression rate trends, they claim that the different modes of heat transfer are dominant at different pool diameters, and that these regions of heat transfer control also correspond closely to the regions of flow regime. At small pool diameters, where the flame is laminar they argue that conduction from the pan is the dominant mode of heat transfer. In the region of large pool size (1 meter or larger in most cases) where the flame is fully turbulent, radiation is the dominant mode. Convection is more or less constant through the whole range of pool diameters, and the region between the two regions of conduction and radiation dominance is where the flow is in transition

to turbulence and the regression rate is at its lowest.

Hottel (1958) supports the reasoning for this explanation of regression rates through a mathematical model of the pool's energy input:

$$\frac{q}{A} = \frac{k_1 \left(T_f - T_B\right)}{d} + U \left(T_f - T_B\right) + \sigma F \left({T_f}^4 - {T_B}^4\right) \left(1 - e^{-\kappa d}\right). \tag{2.3}$$

In this relation, q is the thermal energy to the pool, A is the pools area. The right-hand side of the relation is the sum of three terms. The terms, going from left to right represent mathematical models for heat transfer due to conduction, convection, and radiation. In this part of the relation, d represents the pool's diameter, k_1 is thermal conductivity, T_f and T_B represent the flame and pool temperatures, respectively, U is the heat transfer coefficient for Newton's law of cooling, σ is the Stefan-Boltzman constant, F is a geometric view factor that represents the flames view of the pool surface, and κ is Beer's law extinction coefficient that models the flame opacity with flame thickness. Note that the convection term has no dependence on pool diameter. This supports the assertion that convection heat transfer remains constant through the range of pool diameters. The conduction term contains the pool diameter in its denominator, so when the pool diameter is small the the conduction term will be large and the term becomes smaller as diameter increases. This supports the observation from Blinov and Khudyakov that the conduction dominates heat transfer to the pool for small flames. As pool diameter increases the emissivity of the flame in the radiation term of Equation 2.3 approaches unity and the radiation term grows asymptotically to a maximum value. This confirms that radiation dominates heat transfer to the pool for large diameters, and it confirms that the regression rate will reach a maximum value as pool diameter grows larger.

The dividing diameter between the different heat transfer zones is not uniform for all fuels. For example, Klassen and Gore (1992) found that for toluene fires thermal radiation dominated at diameters as small as 5 to 7 centimeters while for heptane this dominance begins at diameters of 20 to 28 centimeters. For methanol fires of 30 centimeters diameter radiation is important along with convection. The explanation for these differences has to

do with each fuel's ability to radiate heat when burning. This property will be reviewed in more detail later.

It is clear that conduction from the pan loses its dominance rather quickly as diameter increases. Only for pools of 5 centimeters diameter or less is conduction an important effect. Accidental fires considered to be a threat to life or property will certainly be much larger than 5 centimeters. Alger et al. (1979) as well as Klassen and Gore (1992) state that convection is important only in moderately sized fires (20 to 100 centimeters), and that for all flames 1 meter in diameter or larger negligible error is inflicted if convection is ignored, and the error only declines as the size of the pool increases. Since most accidental fires are much larger that 1 meter radiation becomes the most important mode of heat transfer in the study of pool fires. Much more focus will be placed on the effects and properties of radiative heat transfer than for convection or conduction. For simulations of large, turbulent pool fires radiation is the only mode of heat transfer to the pool that needs to be considered. For the simulation of moderately sized pools convection should also be added, especially if the flames are nonluminous (non-soot forming).

As the larger pools have a burn rate determined almost entirely by radiation input, a simple mathematical expression has been developed to give the global burn rate for such fire. Blinov and Khudiakov (1957) as well as Burgess et al. (1961) describe the model. If conduction and convection are neglected then the first two terms on the right-hand side of Equation 2.3 are dropped leaving only the third term. The first part of the radiation input term represents the maximum vaporization rate, and the second part of the term containing the pool diameter dependence is the emissivity of the pool. As stated before the larger the pool the closer the emissivity comes to unity. With these considerations the following simple model for large pool burn rate is proposed:

$$v = v_\infty \left(1 - e^{-\kappa d}\right). \tag{2.4}$$

In this equation, v is the burn rate of the fuel expressed as the liquid regression rate, v_∞ is the maximum regression rate under conditions of a flame emissivity of unity. In both studies by Blinov and Khudyakov and by Burgess et al., the two parameters of Equation

2.4 (v_∞ and κ) are experimentally determined for a number of fuels.

2.5 General Properties of Pool Flames

Blinov and Khudyakov (1957) make some interesting observations about the size and shape of the flames arising from pool fires. They define the flame as the reaction region which separates the zone where there is oxidizer but no fuel from the region where there is fuel but no oxidizer. They comment that the size and shape of the flame are largely dependent on the size of the pool. For the smallest of pool diameters ($\sim$ 10 millimeters or less) the flame has a well-defined conical shape that is fixed (no pulsations.) As the diameter increases the flame begins to exhibit longitudinal pulsations. The height of the flame fluctuates. As diameter further increases the flame begins to break up. Once the pool is larger than 15 centimeters turbulence starts to manifest in the motions of the flame. As to the size of the flame, as the diameter starts to increase from the smallest diameters the ratio of flame height to pool diameter increases sharply throughout the laminar region. As diameter continues to increase the growth in the ratio begins to flatten out. In the turbulent region the ratio remains virtually constant.

Another important factor in the size and shape of the flames of pool fires is the fuel type. Hamins et al. (1994) burn different types of fuel in 30-centimeter pans. They compare the shape and character of flames from heptane, toluene, and methanol. These fuels are chosen because of their different sooting tendencies, and because the shape of the flames produced by these different fuels are markedly different. Hamins et al. describe the shape of methanol flames as unusual compared to that of the other fuels tested. They describe the steady flame as a series of thin, blue sheets starting from the edge of the pan and sweeping inward to the center of the pool in wave-like patterns. In contrast, they describe the heptane and toluene flames as turbulent with a cylindrical shape and flames that are bright yellow in color.

The sooting tendency is the main reason for the difference in flame character described by Hamins et al. Soot formation is an important phenomena that influences the radiative properties of the flame and ultimately the burn rate of the pool as well as the shape and size of the flame. Turns (2000) describes the formation of soot as a gas phase reaction that

occurs for certain fuels at select temperatures at the low point in the flame. He describes the formation and destruction of soot in four steps. The first step is the formation of precursor species or the formation of polycyclic aromatic hydrocarbons (PAH) from the various fuel species. The second step is the particle inception step where small particles of critical size (several thousand atomic mass units) form due to chemical reaction and coagulation. The third step is the surface growth and particle agglomeration of the small soot particles as they are exposed to the fuel-rich environment. The last stage is particle oxidation. As the particles are exposed the oxidation part of the flame, usually in the top regions of the flame, they are destroyed through the oxidation process. Fuels that destroy all of their formed soot are termed "nonsooting." Any fire that is highly efficient in burning the fuel supply and converting it to carbon dioxide will be nonsooting. In pool fires, the flames are not typically so efficient, and sooting frequently occurs. Certain fuels tend to produce more soot that others. Turns ranks the fuels from lowest to highest sooting tendency in this order: alkanes, alkenes, alkynes, then aromatics. Pool fires involving these fuel are soot producing. Alcohol pool fires, in contrast, produce little or no soot, and since thermal radiation originates largely from soot particles alcohol fires generate less radiant heat. Klassen and Gore (1992) report from their measurements of radiation flux from heptane, toluene, and methanol flames in 30-centimeter pans that the radiation heat loss for methanol is about 0.2. The heat loss for heptane and toluene are about 0.3.

Relatively hot soot particles are responsible for the bright, yellow character of luminous flames and are proficient at radiating heat to the surroundings. The more luminous a flame the greater the radiative heat release, including that to the pool surface. Luminous, soot-forming fuels have greater heat feedback the pool surface which causes higher vaporization rates, which in turn leads to larger, more turbulent flames. Also, because luminous flames lose more heat to thermal radiation, their flame temperatures are lower than that of nonluminous flames as is confirmed experimentally by Rasbash et al. (1956) where they report that the flame temperature for the alcohol mixture is about 1200^{O} C. The same measurements for kerosine and petrol are about 200^{O} C lower than that of the alcohol flame, and the temperature for benzole, a mixture of aromatics which has a higher sooting tendency, is about 300^{O} C lower than that of the alcohol flame. Also, Alger et al.

(1979) report from their 3-meter pool experiments that the measured flame temperature for methanol is about 200^{o} C higher than that of JP-5.

In their work with 30-centimeter fires Rasbash et al. (1956) make some relevant comments on the formation of the flame shapes for different types of fuels. At the beginning of the burn a thin sheet of flame originates from the pan's edge and slopes downward to touch the surface of the pool before sloping upward to form a cylindrical shape with diameter smaller than that of the pan. As the burn process continues, fuel evaporates from the surface at an increasing rate which pushes the flames away from the pool surface and forms a fuel rich zone between the pool surface and the flame. Figure 2.3 shows a sketch of the shape of flames near the pool at different pool fire stages as described by Rasbash et al.

As shown in Figure 2.3(A), at the start of the burning process the flames can touch a substantial portion of the pool surface. For the alcohols tested in their study Rasbash et al. found that this flame shape held for the duration of the burn. Figures 2.3(B) and 2.3(C) illustrate what happens to more luminous flames as the burning continues. During the steady burning period the flames fluctuate between the shapes shown in B and C of Figure 2.3. This pattern is consistent with the behavior of the different types of fuels. The nonluminous alcohol flames radiate less heat to the pool surface and generate less vaporization of the fuel. The influx of fuel is not enough to push the flames far from the pool surface. However the luminous flames produced by the kersosine, petrol, and especially benzole generate enough fuel vaporization to form a larger vapor dome over the surface and push the flames much higher. The formation of the vapor dome over the pool surface is an important phenomenon as it can affect the volatility of the liquid fuel, the convection heat transfer to the pool surface, as well as the radiation heat transfer to the pool surface.

2.6 General Properties of the Liquid Pool

In their measurements of pool regression rate with pool fires Burgess et al. (1961) note that after the fire starts there is a burning-in period where the burn rate accelerates before reaching a steady burning period where the regression of the liquid level is constant. This

behavior is consistent with other observations of burning liquid fuels. As the flames grow they heat the pool, and as the pool warms it gradually vaporizes more fuel until a steady state is reached. Chatris et al. (2001) also note this transient behavior in their experiments. They describe the transient burning process in three steps as follows: The first period is a transitory warm up phase. In this phase the flames spread over the fuel surface, and then the flames gradually increase in size as the pool warms. They note that for gasoline this stage lasts 40 to 50 seconds, and for diesel oil it takes about 60 seconds. The second period is a stationary period characterized by a fully developed fire with relatively constant burn rates. This stage ends when the fuel depletes, so the length of time for the stationary phase depends on the amount of fuel in the pool. The last stage is another transitory period where the flame size and burn rates gradually diminish until the fire extinguishes.

A very important property of the liquid in pool fires is the temperature of the liquid. Blinov and Khudyakov (1957), Hottel (1958), Rasbash et al. (1956) all note that when a pool fire ignites and the flames heat the liquid pool the surface temperature eventually reaches a steady value that is usually just below the boiling point of the liquid. This is found to be true of single component fuels and mostly true of fuel mixtures. From their measurements of liquid pool temperature Blinov and Khudyakov find that the temperature profiles with liquid depth follow a similar pattern for all measured fuels. Figure 2.4 is a qualitative sketch of the common temperature profile.

Blinov and Khudiakov also propose a simple mathematical model for the liquid temperature with depth:

$$\frac{(T(z) - T_O)}{T_S - T_O} = e^{-kz}. \tag{2.5}$$

In Equation 2.5, T(z) is the liquid temperature as a function of liquid depth, z (z = 0 corresponds to the pool surface.) T_O is the nominal temperature or temperature of the surroundings, and T_S is the pool surface temperature. The last symbol, k, is a parameter that determines the shape of the temperature curve. Blinov and Khudyakov find that this parameter varies widely for different fuels. They also find this parameter varies with other physical properties such as the material of the pan in which the fire burns, the diameter

of the pan, and the wind speed. They also find that k can vary with time. The important detail is not so much the specific mathematical model or its parameter values, but the shape of the curve itself. Examination of Figure 2.4 shows that the heating of the liquid occurs in a small ($\sim$ 1 centimeter) layer near the the surface. Blinov and Khudyakov largely account for the shape of the temperature through the thermal diffusivity of the liquid. Since the thermal diffusivities of hydrocarbon liquids are relatively small the temperature profile would be expected to drop sharply past the pool surface. Also, as mentioned already, Alger et al. (1979) find that the thermal radiation to the pool surface penetrates to a small depth of the liquid pool, thus heating only that part of the pool.

Blinov and Khudyakov also report on the horizontal variations of the liquid temperature. For 25-centimeter pool fires they find that the temperature is highest near the edge of the pool and decreases toward the center. This is expected because the flames come closest to the pool near the edge of the pan. Both convective and radiative heat fluxes would be high at the edge of the pool leading to higher temperatures there. For larger pools it is found that radiative heat flux becomes large at the center of the pool which leads to higher temperatures in that location as well. Blanchat et al. (2006) demonstrate this heat flux pattern with measurements they take from 8-meter pool fires.

Another important property of the liquid pools is the composition of the fuel itself. Blinov and Khudyakov (1957) describe some of the important effects of composition in their report. Rasbash et al. (1956) report that for each of the mixtures burned in their experiments the measured boiling point of each fuel changed during the course of the experiment. This observation is common for experiments involving fuel mixtures, and it is an expected trend since mixtures tend to vaporize the light components first. As they deplete the lighter components from the liquid phase, the boiling point of the liquid rises. The magnitude of change in the boiling point during the pool fire depends on the components present. Some mixtures contain hundreds of components with a wide range of boiling points. Other mixture's components may have a relatively small range of boiling points. Rasbash et al. report that for an alcohol mixture the boiling point changed from 77^o C to 79^o C. Their report for benzole shows slightly larger boiling point change from 79^o C to 84^o C. Their data for kerosine and petrol demonstration much larger boiling point

changes (155^o C to 277^o C for kerosine and 30^o C to 200^o for petrol.)

Besides the change in boiling point, Blinov and Khudyakov observe different mixing patterns in fuel mixtures as they burn. In some mixtures they note that the top layer of the pool is well mixed giving a uniform composition to the liquid in that layer. This is ascribed to mixing that occurs when the surface liquid is cooled somewhat by the process of evaporation. The cooled liquid is more dense causing it to mix into the immediate lower layers and causing the mixing pattern in the heated layer of the liquid pool. In other cases it was noted that mixing occurred throughout the entire liquid pool. This happens for mixtures where the density differences between the various components is large enough to cause circulation throughout the liquid. The last type of liquid behavior is the formation of what Blinov and Khudyakov call the homothermal layer. As already mentioned here, the surface temperature of burning liquids tends to come to a steady value that is slightly below the boiling point of the liquid. For some mixtures this is not true. The homothermal layer is a well mixed zone where boiling of lighter components occurs. Under certain conditions if the layer grows in size, boil over occurs. Blinov and Khudiakov describe the formation and propagation of the homothermal layer in their report. As the fuel mixture heats up, the temperature of the liquid is less than the temperature of the pan wall. If the pan wall temperature rises higher than the liquid temperature then lighter components at the bottom of the layer begin to boil near the wall. The boiling causes vigorous mixing in the layer which can convect heat to lower layers in the liquid. As it does so and as the temperature of the pan gets hotter at lower locations the homothermal layer spreads and boiling increases. If the fuel is situated on top of a layer of water the heat from the fuel will heat the water. Depending on the conditions of the water (purity and presence of nucleate forming material) the water can heat to its boiling point and even superheat beyond its boiling point. When the water can form steam bubbles, it can do so quite violently. The expulsion of water vapor from the pool can project burning fuel for large distances creating a severe safety hazard. As noted above, not all liquids can form homothermal layers or boil over. First, this can only occur with mixtures. Also, for the boiling to happen the mixture must have components with a sufficiently low boiling point ($\sim 100^o$ C). The presence of water in many fuels promotes this behavior. Windy

conditions can promote this phenomenon since the winds can deflect the flames in such a way as to immerse the pan and speed the process of heating at the pan wall. Water cooling of experimental pool pan when properly applied can suppress the heating of the pan and prevent boil over.

2.7 Properties of Pool Fire Radiation and Convection

As previously stated, most accidental fires are large and turbulent. This means that the dominant mode of heat transfer to the pools and thus the most important property to influence the pool vaporization is thermal radiation. This section, therefore, will address mostly properties of thermal radiation and only spend a minimal effort on convection.

Thermal radiation is one of the three basic modes of heat transfer. According to Incropera and Dewitt (2002), unlike the modes of conduction and convection, the propagation of radiant heat does not require the presence of matter in the medium of propagation. Radiation is emitted from all matter though electromagnetic waves that are generated by the oscillations and transitions of electrons present in all matter. The molecular oscillations and electronic transitions are maintained by the internal energy and thus the temperature of matter. Thermal radiation is transmitted by electromagnetic waves of a certain range of wavelengths. Figure 2.5 is an illustration of the electromagnetic (EM) spectrum with the range of radiative heat transfer labeled.

As is evident in Figure 2.5, radiative heat waves cover the visible range, a substantial portion of the ultraviolet range, and most of the infrared range. The wavelength of the emitted radiation depends on the molecular properties of the material from which it is emitted. Different molecular vibrations and oscillations as well as electronic transitions have different energies and emit waves at different wavelengths.

In the pool fire system, the thermal radiation is emitted from two sources. The two sources are radiation emitted by optically active gases in the flame region. These include unreacted hydrocarbons, carbon dioxide, water, carbon monoxide, etc. The second source of radiant heat is emitted from hot soot particles formed in the combustion process. Alger et al. (1979) report that radiation emitted from gas species exhibits wavelengths which are generally less that 1 micrometer, while the radiation emitted from soot has wavelengths

mostly larger that 1 micrometer. Since nonluminous flames, such as those that arise from alcohol fires have little or no soot, their thermal radiation arises largely from molecular emission with smaller wavelengths. This explains why alcohol flames are often blue in color (blue light has wavelength between 0.44 and 0.49 micrometers.) Sibulkin (1979) reports that the size of the pool also effects which types of radiation are more important. He finds that for pool fires of 10 centimeters or less in diameter, molecular radiation dominates over soot radiation. As the size of the pool increases the influence of soot-generated radiation becomes more important. For large fires both sources of radiation are important.

As mentioned before, Klassen and Gore (1992) perform pool experiments with methanol, heptane, and toluene to compare flames from nonsooting, moderately-sooting, and heavily-sooting fuels. They find that radiative heat loss for the alcohol flames is about 0.2, and that the same property for heptane and toluene is about 0.3. Radiative heat loss is defined as the amount of heat released as radiation from the combustion reaction divided by the heat of combustion of the fuel. Klassen and Gore note that despite the difference in soot-producing tendency between heptane and toluene, their radiative heat loss was the same. They explain that the reason for this is that while toluene pool fires may generate a great deal of soot, a large fraction of the generated soot particles are relatively cold and do not contribute significantly to the emitted radiation.

Another radiation related phenomenon investigated by Klassen and Gore is the distribution of incident radiation on the pool surface. According to them, for 30-centimeter pool fires of toluene and heptane, the feedback flux is already dominated by radiation. For methanol fuel at the same pool size, convection still plays an important role in heat feedback. Their measurements of radiative heat feedback show that radiation feedback is strongest at the pool center for all three fuels tested. For the luminous flames, toluene and heptane, the strength of the radiation flux remained relatively constant moving out from the center to the edge of the pan. In the case of heptane, there is a slight decline. In contrast, for methanol the strength of the radiative flux decreased significantly toward the edge of the pool. Although radiation feedback is strongest at the center the authors find that the overall heat flux is constant across the surface. This is due to convective feedback being strongest at the edge of the pool. For moderate to large pools the radiative feedback

is strongest in the center while the convective feedback is strongest at the edge where the flames comes closest to the pool surface.

Another interesting feature of the radiative properties of pool fires is the absorptivity and transmissivity of the flame environment to thermal radiation. This particular property is of interest because it is important to know where the bulk of the radiative flux that is incident on the pool surface originates. In their measurements of the optical properties of the soot-forming flames Klassen and Gore (1992) find that for toluene, the more soot-forming of the two fuels, the resulting flames are optically thick with transmittances less than 30% throughout the entire flame and as low as 5% in some regions. For heptane, they find the flames to be optically thin with transmittances of 70% or greater throughout the entire flame region. The effect of the flame's optical properties on the pool are that for optically thick toluene the incident radiation at the pool originates from the flame within a height above the pool equal to 0.5 times the pool diameter. For optically thin heptane the incident radiation originates within a height above the pool equal to 3.5 times the pool diameter. These properties are important because for pool fires deflected by high crosswinds the incident angles of radiative feedback would be changed particularly for optically thin flames. Radiation feedback with lower incident angles would have a higher reflectivity, and this would significantly affect the energy balance of the pool.

The liquid pool itself also has optical properties which are of great import. When a wave of thermal radiation strikes the surface of the pool it is not entirely absorbed into the pool. Some of the energy is transmitted into the pool, and some of it is reflected off the pool surface back to the surrounding domain. Of the radiation that is transmitted into the pool some will be absorbed more quickly, and some will transmit further before it is absorbed. Since the pool surface has temperature it will radiate its own heat in a process called re-radiation. Figure 2.6 illustrates these phenomena.

The simplest of these phenomena to model is surface re-radiation. Both Hamins et al. (1994) and Brown and Vembe (2006) use the Stefan-Boltzman law:

$$q^{''}_{re-rad} = \epsilon\sigma(T_s)^4. \tag{2.6}$$

In this relation, ϵ represents the emissivity of the pool surface, σ is the Stefan-Boltzman constant, and T_S is the pool surface temperature. The value of the emissivity is unity for the ideal, blackbody surface, and Hamins et al. make this assumption. Brown and Vembe assume Kirchoff's approximation that the emissivity of the surface is equal to its absorption, and they assume that the absorption is 1 minus the reflectivity.

For the absorption of the radiation that transmits through the pool surface, it has already been mentioned earlier that the bulk of the radiation is absorbed within a depth of the pool that is approximately 1 centimeter. The absorption of light in an absorbing medium is described by the Beer-Lambert law. This law is expressed in terms of transmittance:

$$T(z) = \frac{I(z)}{I_O} = e^{-\alpha z}. \tag{2.7}$$

Here, T is the transmittance of light, and it is expressed as a function of path length z which in this physical case is the depth of the pool below the surface. The transmittance is defined as the ratio of the intensity of the light $I(z)$, at some depth z, divided by the incident intensity of the light I_O. The mathematical form of the Beer-Lambert on the far right side of Equation 2.7 shows that the intensity of the radiation drops exponentially with pool depth. This is consistent with the idea already expressed that the thermal radiation does not penetrate far into the liquid depth. Also, recall that the steady temperature profile as a function of pool depth also drops in a similar manner as seen in Figure 2.4. The quantity α in Equation 2.7 is the absorption coefficient and is expressed:

$$\alpha = \frac{4\pi\kappa}{\lambda}. \tag{2.8}$$

In this expression λ is the wavelength of the radiation, and κ is the imaginary part of the refractive index of the medium (in this case the liquid fuel), also sometimes called the absorption coefficient. The value for the absorption coefficient varies from liquid to liquid. Examination of steady temperature profiles for pool fire show that the heated regions of the pool occur within about 3 centimeters of the pool surface, which suggests that the coefficient may not vary widely among different liquid fuels.

Modeling surface reflection requires the application of the Fresnel equations. These equations describe the fraction of intensity of light that is reflected, commonly called the reflectance, as it passes the interface between one medium of refractive index n_1, and another medium of refractive index n_2. The model is described by two equations:

$$R_s = \left[\frac{n_1 \cos\theta_i - n_2 \sqrt{1 - \left(\frac{n_1}{n_2} \sin\theta_i\right)^2}}{n_1 \cos\theta_i + n_2 \sqrt{1 - \left(\frac{n_1}{n_2} \sin\theta_i\right)^2}} \right]^2 , \tag{2.9}$$

and

$$Rp = \left[\frac{n_1 \sqrt{1 - \left(\frac{n_1}{n_2} \sin\theta_i\right)^2} - n_2 \cos\theta_i}{n_1 \sqrt{1 - \left(\frac{n_1}{n_2} \sin\theta_i\right)^2} + n_2 \cos\theta_i} \right]^2 . \tag{2.10}$$

The Fresnel relations apply to light waves traveling through nonmagnetic media. Examination of Equations 2.9 and 2.10 show that the reflectivities are defined in terms of the real part of the refractive indexes of the media, n_1 and n_2, and the angle of incident radiation, θ_i. The reflectivity defined in Equation 2.9 is for light polarized with the electric field perpendicular to the horizontal plane depicted in Figure 2.7. The reflectivity defined in Equation 2.10 is for light polarized with the electric field along the plane in Figure 2.7. Figure 2.7 is an illustration of the physical phenomena associated with the reflection and refraction of light as just described.

For light that is an equal mix of both polarizations, the following relation can be used to get the overall reflectivity:

$$R = \frac{(R_s + R_p)}{2}. \tag{2.11}$$

Since reflectivity depends on the real part of the refractive index of the medium, reflectivity can be potentially different for each fuel. Alger et al. (1979) comment that most hydrocarbon fuels of interest have refractive indexes ranging between 1.4 and 2.0. Hamins et al.

(1994) find that the Fresnel relations describe pool surface reflectivity well, and they find that a refractive index of 1.5 fits the reflectivity profile as a function of incident angle for water, heptane, methanol, and toluene quite well. To further illustrate this point and to demonstrate how reflectivity changes with incident angle, Figure 2.8 is provided. Figure 2.8 shows the reflectivity of incident radiation on a pool surface for all possible incident angles and for refractive index ranging from 1.3 to 2.0 using the Fresnel relations (Equations 2.9, 2.10, and 2.11.) Note in Figure 2.8 that reflectivity is lowest when the incident radiation approaches vertically (incident angle of 0^o), and that reflectivity is highest at shallow angles (incident angle approaches 90^o). The definition of incident angle used here is demonstrated in Figure 2.7. Also, note from Figure 2.8 that the difference in reflectivity among different refractive indexes is small especially at high angles.

To get the overall reflectivity of the incident thermal radiation at the pool surface requires information on the angular distribution of the incident radiation. Alger et al. (1979) estimate the average reflectivity to be between 0.05 and 0.10. Hamins et al. (1994) use angular data and an integration scheme with the reflectivities calculated from the Fresnel relations to estimate the average reflectivity to be between 0.05 and 0.08 for the pool fires they test.

Although not as important as radiation, convection does contribute some input to the pool surface and is important in the simulation of moderately sized pool fires. As defined by Incropera and Dewitt (2002), convection is the mode of heat transfer with two mechanisms: random molecular motion or thermal diffusion, and bulk flow motion from a fluid. There are different modes of convective heat transfer (forced and free convection for example), but regardless of the mode convection is generally modeled using a relation called Newton's law of cooling as is given in the following equation:

$$q''_{conv} = h\left(T_S - T_\infty\right). \tag{2.12}$$

Here, T_S is the temperature of the surface of interest. In this case the surface is that of the liquid pool. T_∞ is the temperature of the surrounding fluid. In a pool fire the temperature of the gas immediately above the pool rapidly rises in temperature. The

greater the difference in temperature between the pool surface and the adjacent fluid the greater the convective heat flux to the pool. In pool fires where the flames are situated close to the pool surface the convection heat transfer is greatest. As mentioned before, this tends to occur near the edge of the pool at the beginning stages of the fire and in pools with lower vaporization rates. In Equation 2.12, h is the heat transfer coefficient, and its magnitude depends on the conditions of the interface between the surface and the fluid as well as the geometry and fluid flow conditions of the system. In general free or buoyant convection system give lower heat transfer coefficients than systems with forced flow or phase change. The presence of significant mass flux at the surface/fluid boundary can also enhance the rate of convective heat transfer.

2.8 Previous Liquid Pool Models

As this is not the first attempt to model the liquid pool for the purpose of predicting burn rates in pool fires, it is appropriate to review three past models.

2.8.1 Model 1: Prasad et al. (1999)

Prasad et al. (1999) make the first attempt to develop a coupled gas phase/liquid phase model to more accurately predict the behavior of small (diameter $\sim$ 1 centimeter) methanol pool fires with the objective of predicting the pool's burn rate. The authors develop a two-dimensional, differential conservation model for both the gas and the liquid phases of a pool fire system. For the methanol reaction model they use is a simple one-step reaction mechanism. The radiation model they use assumes optically thin conditions which is valid for the modeling of nonluminous flames. The gas phase conservation equations are the DNS model and are therefore only tractable for laminar flame modeling. The liquid phase conservation model is described here:

$$\frac{\partial\left(\rho_f C_f T\right)}{\partial t} + \frac{\partial\left(\rho_f C_f T r_b\right)}{\partial y} = \lambda_f \left(\frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial x^2}\right) + Q_{abs}. \tag{2.13}$$

In this model, Q_{abs} is the thermal input to the pool surface from radiation, convection and conduction, and r_b is the liquid regression rate. The x-coordinate is the spatial direction horizontal or parallel to the pool surface, and the y-coordinate is the direction vertical

or perpendicular to the pool surface. The accumulation term on the left hand side and the molecular diffusion term on the right hand side of Equation 2.13 are straightforward enough. What is perhaps not so intuitive is the convection term on the left hand side. In their formulation of the model, the authors view the liquid/gas interface as fixed in their coordinate system, so as the liquid regresses instead of the surface moving downward, the entire liquid pool below the surface moves upward at a rate equal to the pool's rate of regression. In their solution scheme, the authors do not attempt to predict the regression rate directly. Rather, they guess values for the regression rate and solve for the temperature over the whole domain. They repeat the process iteratively until a certain condition is met. The conditions is the equivalence of two pressures. The first is the partial pressure of the fuel expressed here:

$$P_{vap} = X_{vap}P. \tag{2.14}$$

Here, P_{vap} is the partial pressure of the fuel, P is the total system pressure and X_{vap} is the mole fraction of the fuel based on the gas phase calculations. The other pressure is the equilibrium vapor pressure of the fuel computed from the Clausius-Clapeyron equation:

$$P_{cc} = P_1 e^{\left(\frac{-\Delta H_v}{R}\left(\frac{1}{T_{cc}} - \frac{1}{T_1}\right)\right)}. \tag{2.15}$$

When the computed partial pressure of the fuel is equal to its Clausius-Clapeyron vapor pressure the iterative process is done for that time step.

The authors assume for their model that horizontal mixing within the pool is negligible. Their model for radiation includes the effects of absorption into the surface layers of the pool. The authors do not explicit state in their paper whether they include the effects of surface reflection or re-radiation, but since the range of pool fires sizes modeled are in the conduction dominated zone, these effects are not likely important. The liquid regression rate predicted by their model for a small liquid methanol fire is 3.4×10^{-5} m/sec. The regression rate measured by Akita and Yumoto (1965) is 5.0×10^{-5} m/sec (32% deviation between prediction and experiment.)

2.8.2 Model 2: Novozhilov and Koseki (2004)

Another coupled gas/liquid phase model is proposed by Novozhilov and Koseki (2004). For the gas phase the authors incorporate a single generic chemical reaction for hydrocarbons as their kinetic model. The radiation model used in their code tracks the change in radiative intensities with path distance due to absorption and emission of radiation by gas and soot particles using a method called the discrete transfer method described by Lockwood and Shah (1981). The authors incorporate an empirical correlation for the absorption properties of optically active gases developed by Fletcher et al. (1994), and they assume a constant soot conversion factor for the formation of soot. Each fuel has a different soot conversion factor. The computational domain for the liquid and gas is three-dimensional, but the conservative equations given by the authors suggests that horizontal heat conduction and mixing are neglected. The authors model the energy balance with the liquid pool with a simple conduction model:

$$\frac{\partial T}{\partial t} = k_f \frac{\partial^2 T}{\partial z^2}. \tag{2.16}$$

The authors also take into account the decrease in the fuel depth as the liquid burning process proceeds. The entire horizontal surface regresses at the same rate, and the rate is determined by the universal pool burn rate which must be determined from the sum of the local burn rates over the surface of the pool. The authors acknowledge the importance of thermal input to the pool as deterministic of the burn rate. They treat the thermal input to the pool as one of two necessary boundary conditions for the liquid model, and it is expressed as an energy balance across the pool surface:

$$k_f \frac{\partial T}{\partial z} = Q_f - m'' H_{fg}. \tag{2.17}$$

In this relation ,Q_f is the heat input to the pool, and it includes incident radiation, convection, and re-radiation from the pool. H_{fg} is the heat of vaporization for the fuel. The authors do not mention surface reflection of radiation in their paper. Although the authors describe the solution of the conservative equations as iterative they do not describe

the procedure in any more detail. The other boundary condition is that the pressure of fuel species at the liquid/gas interface is equal to the fuel's vapor pressure for the temperature of the liquid at the surface. Using this condition the author calculate the fraction of fuel species at the interface, and they use it with a mass transfer model to calculate the fuel mass burn rate:

$$m'' = h_m \left(Y_{F,\,s} - Y_{F,\,\infty} \right). \tag{2.18}$$

Here, $Y_{F,\,s}$ is the fraction of fuel at the gas/liquid interface, and $Y_{F,\,\infty}$ is the fraction of fuel in the gas phase. The mass transfer coefficient, h_m is computed under the assumption that the heat/mass transfer analogy applies. The mass transfer coefficient is obtained from the following relationship to the heat transfer coefficient, h, described by Incropera and Dewitt (2002):

$$\frac{h}{h_m} = \rho_g C_{p,\,g} Le^{1-n}. \tag{2.19}$$

In Equation 2.19, Le is the Lewis number. For the parameter n, it is recommended that a value of $\frac{1}{3}$ be used. The authors estimate the heat transfer coefficient by use of a computer code that applies the conventional wall-function correlations developed for parabolic flows by Patankar and Spaulding (1972).

The authors test their model with simulations of pool fires of size ranging from 30 centimeters to 1 meter using fuels such as heptane, toluene, and methanol. The authors claim good agreement between their predictions of radiation feedback to the pool surface within experimental error. They also report an average 17% error in the predicted fuel burn rates with the worst discrepancy occurring with heavily sooting flames. Among the experiments used to validate the author's model are the pool fire tests conducted by Klassen and Gore (1992). It is also worth noting that the authors did not apply their model to the prediction of pool fires under conditions of crosswinds.

2.8.3 Model 3: Brown and Vembe (2006)

The last pool model discussed here is developed by Brown and Vembe (2006). They develop this model to be used in conjunction with existing CFD fire codes in order to predict the burn rate of the fuel to the gas domain. The authors allow the pool surface temperature to vary horizontally. Unlike the model developed by Novozhilov and Koseki (2004), the vertical aspect of the pool (the pool's depth) is represented in the computational domain with only 1 node. This implies that there is no modeling of the pool's temperature profile with depth. For the conservation of energy at the pool, incident radiation, reflected radiation, re-radiation, and convection are taken into account:

$$q^{''}_{total,pool} = \alpha q^{''}_{incident} - \alpha\sigma T^4_{pool} + q^{''}_{convection}. \tag{2.20}$$

As mentioned earlier, in their energy balance (Equation 2.20) the emissivity of the pool is assumed to be the same as the absorptivity. Incident radiation and convection are computed from the gas-phase portion of the fire code. The authors note that not all heat absorbed by the surface of the pool is used to evaporate fuel (also called latent heat.) Some is also used to heat the liquid (also called sensible heat.) In place of a mass/heat transfer correlation to predict latent heat, the authors develop a simple, empirical correlation based from the heating of nonane (C9H12) to predict the fraction of sensible heat to the pool as a function of pool temperature:

$$f_{sensible} = max\left(min\left(\frac{\left(T_b - T_{pool}\right)}{\Delta T_b}, f_{max}\right), 0\right). \tag{2.21}$$

Here, T_b is the boiling point of the fuel, and it will vary from fuel to fuel. The authors assign values to the parameters f_{max} and ΔT_b of 0.7 and 125^o C, respectively. With definitions from Equations 2.20 and 2.21, the author's model for pool temperature, liquid height, and fuel burn rate can be given:

$$\frac{dT_{pool}}{dt} = \frac{q^{''}_{total,pool}\, f_{sensible}}{h\rho C_p}, \tag{2.22}$$

$$\frac{dh}{dt} = \frac{q^{''}{}_{total,\,pool}\left(1 - f_{sensible}\right)}{H_{evap}\,\rho}, \tag{2.23}$$

and

$$\frac{dm}{dt} = \frac{q^{''}{}_{total,\,pool}\,A\left(1 - f_{sensible}\right)}{H_{evap}}. \tag{2.24}$$

As is evident from the model, the authors take into account the drop in pool liquid level and the variation of temperature and mass burn rate over the pool surface. The also assume that horizontal heat flux within the pool is negligible. Instead of modeling a three-dimensional domain, the authors implement an instance of the above described one-dimensional (in time only, no spatial variation) model at each section of the pool surface. This implies that a different liquid height can exist at the different sections across the pool surface, and as already stated the various instances of the pool model across the surface do not interact with each other including the transfer of heat from pool element to pool element. Four experimental pool fires are used to validate the code. Three of the tests involve the burning of jet fuel (JP-8 and Jet-A) in pans ranging from 2 to 3 meters in diameter (one of the pans is square with side length 2.7 meters.) The fourth test is a methanol fire, 2 meters in diameter. Some of the experiments use crosswind. Each test is simulated using the author's model at using different mesh resolutions in their computational domains. The authors find that the predicted fuel burn rate was within $\pm 50\%$ of the measured rate with two of the four simulation's results being reported well within experimental error. The authors report that the fuel burn rate results are particularly sensitive to the input radiation values. They recommend more attention be given to the sensible heat fraction model to improve their model.

2.9 Methods of Model Verification, Validation, and Error Quantification

From the sources of information described in the previous sections of this chapter a suitable mathematical model of the liquid pool can formulated, but whatever form that model takes, in order for that model to be of any use there must be a reasonable effort

made to verify and validate the model's performance. This includes quantification of the error in the model's results. Figure 2.9 illustrates the different phases of computational model development, and it shows how verification and validation fit into the process.

As mentioned already, verification is the process of determining if the computer implementation of the model adequately represents the mathematical model, and validation is the process of determining if the computer implementation of the model represents the physical system of interest. A more concise way of defining these terms is that verification is determining if the equations are being solved correctly, and validation is determining if the right equations are being solved. Certain methods have been developed over the years to perform code verification and validation. Methods have also been developed for the quantification of the error of a particular model's output.

2.9.1 Methods of Verification

For the process of verification, Roache (1998) describes primarily two methods to determine the error of a computational model. The first method is called the Method of Manufactured Solutions (MMS), and the second method is computation of the Grid Convergence Index (GCI) which is based heavily on the Richardson extrapolation technique developed by Richardson (1910 and 1927).

It is common in the computational sciences to model physical systems using differential equations. The conservation relations that form the Navier-Stokes equations are an example of this as is seen in the previous chapter (Equations 1.1, 1.2, and 1.3.) The common schemes applied to solve these types of equations convert the differential equations to more easily solved algebraic equations using discrete approximations for continuum equations. These approximations come at the cost of inflicting error in the solution process. An example of this is the use of discrete approximations for derivative terms found in differential equations. For example, if the differential relation contains a spatial second derivative the following approximation is often used:

$$\frac{\partial^2 U}{\partial x^2} = \frac{U_i - 1 - 2U_i + U_i + 1}{\Delta x^2} + O\left(\Delta x^2\right). \tag{2.25}$$

The approximation shown in Equation 2.25 and others similar to it can be derived from

Taylor series expansions for any function U. The quantity, U, is a generic symbol for any physical quantity of interest that is modeled in a differential equation. In this example, x is used as the independent variable, but discretizations for any spatial variable as well as the temporal variable are used in these types of derivative approximations. The first term on the right-hand side of Equation 2.25 is the approximation to the derivative shown on the equation's left-hand side. The second term on the right-hand side represents the error inflicted through the application of this discrete approximation. As these approximations are derived from Taylor series there are an infinite number of terms involved that could be used in the definition of the derivative, but only the first term is used to approximate the derivative. This means that the remaining terms resulting from the Taylor series represent the error in the approximation. The first of these terms is used to express the bulk of the error since it usually, but not always, represents the largest of the contributions to the error. In the example of Equation 2.25 the largest error term contains a dependence on the square of the spatial step. The term $''O\left(\Delta x^2\right)''$ means that the bulk of the error is on the order of the spatial step size squared. A more mathematically precise way of saying this is:

$$Error \approx O\left(\Delta x^2\right) = k\Delta x^2. \tag{2.26}$$

n this relation, k is a constant that depends on the function being evaluated and its derivatives. In this example the error depends largely on the spatial step to the second power. Because of this, this type of approximation is said to be second-order accurate with respective to the spatial step size. Smaller discretization error generally results from smaller step sizes. There are a number of approximations that exist to approximate the various derivative terms in differential equations, and each of them has a different order of discretization error. This is also called the order of convergence. Higher order approximations give smaller error, but they can be more difficult to implement. Discretization error is also called truncation error as it results from the truncation of Taylor series.

As already mentioned, computer bugs, discretization error, iterative error, and machine round off error are all sources of error that can arise in computational simulation and code

development and are the subject of study in code verification. The Method of Manufactured Solutions is a useful way to determine the extent of these errors. The idea is that in the processes of solving of a differential model, a solution function is sought. If the function were already known exactly, then the solution produced by the code could be compared to the exact solution to quantify the error. Unfortunately, such exact solutions are not generally available, especially for nontrivial systems. The MMS seeks to manufacture a simple, analytic solution to the differential equation. This is accomplished by adding a source term to the differential equation. For example, the following is a simple differential equation that describes one-dimensional, transient heat conduction:

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2}. \tag{2.27}$$

If a code were developed to solve this equation for a given geometry and it was desired to see if the code was working properly and if it was desired to quantify the error in the code a simple function could be manufactured for T. To be useful, the manufactured function would need to depend on t and x, and it would need to be sufficiently differentiable to produce the derivatives found in Equation 2.27. The model would then need to be modified to include the MMS source term as show here:

$$\left(\frac{\partial T}{\partial t}\right)_{discret} = \left(\alpha \frac{\partial^2 T}{\partial x^2}\right)_{discret} + q_{mms}. \tag{2.28}$$

Here, q_{mms} is the source term for MMS. Also note that the derivative terms are given the *discret* subscript to indicate that these terms are computed using a discrete approximation such as the one described by Equation 2.25. If the MMS source term is properly computed the code will generate the manufactured solution with some error. The source term is found by taking Equation 2.28 and solving for the source term and switching the approximated derivative terms with analytically determined derivatives from the manufactured solution as shown here:

$$q_{mms} = \left(\frac{\partial T}{\partial t}\right)_{analytic} - \left(\alpha \frac{\partial^2 T}{\partial x^2}\right)_{analytic}. \tag{2.29}$$

The chosen manufactured solution need not have any physical relation to the system. It only needs to be a simple, analytic function that can produce the needed derivative terms. The manufactured solution provides an exact solution that can be compared to the solution generated by the code, and errors can be evaluated. Inspection of the solution generated by the code can reveal if there are hidden programmer errors. Grid convergence analysis can reveal if the order of the discretization error from the MMS analysis is what it should be. If a second order approximation with respect to the spatial variable is used in the algorithm and a first order error is observed, it could indicate a bug in the code or a subtle feature of the algorithm itself. The point is that in this type of exercise information about a code's performance is revealed, and code performance could possibly be improved.

Even with MMS, determination of the actual order of the discretization error requires grid convergence analysis. This type of analysis requires the same simulation to be executed using different spatial and temporal mesh resolutions. A common approach is to run a simulation at a relatively coarse resolution and then run subsequent simulations with meshes that successively double the resolution (the step size is cut in half, or their are twice as many nodes in a given spatial direction or in time.) By comparing similar outputs from simulations of different mesh resolution the order of the discretization error can be measured, and the error can be estimated. Methods for this type of anaylsis are based on the Richardson extrapolation. Richardson (1910) examines the Taylor series upon which differential approximations are based and formulates a way to eliminate the largest error terms using solutions from simulations of different resolution thus producing an approximation of much smaller error. This technique is demonstrated in this equation:

$$f_{exact} = f_1 + \frac{f_1 - f_2}{r^p - 1} \tag{2.30}$$

In this relation, f is the function value, where the subscript $''exact''$ denotes the exact solution, the subscript 1 indicates function value generated from simulation with the finer mesh resolution, and the subscript 2 denotes the solution estimated from the coarser resolution. The value r is the mesh refinement ratio, and a value of $r = 2$ would indicate that the mesh resolution is doubled from simulation 2 to simulation 1. The quantity p is

the order of discretization error. For example, for a second order method, the value of p is 2. Equation 2.30 provides a means to get a better estimation of a quantity using two mesh resolutions, but it also provides the means to estimate the error of discretization, and thus provides means to quantify the error. Roache (1998) defines an estimated fractional error, E_1 based on grid refinement and Richardson extrapolation:

$$E_1 = \frac{\epsilon}{r^p - 1}, \tag{2.31}$$

where

$$\epsilon = \frac{f_2 - f_1}{f_1}. \tag{2.32}$$

Roache calls this quantity an ″ordered error estimator,″ but he does not consider this to be a suitable conservative standard for error that is comparable to the error standards applied by experimentalists, and this error estimator is certainly not suitable as a bound on the discretization error. In determining a suitable method for determining error Roache states, ″What is generally sought in engineering calculations is not a true ″error bound″ but just an ″error band,″ i.e., a tolerance on the accuracy of the solution which may in fact be exceeded, but in which the reader/user can have some practical level of confidence.″ To achieve this error band, Roache proposes the Grid Convergence Index (GCI):

$$GCI[finegrid] = F_S \frac{|\epsilon|}{r^p - 1},\ F_S = 3. \tag{2.33}$$

The design of the GCI is to relate the error expressed as ϵ (as defined in Equation 2.32) from any convergence study (meaning any value of p and r) to the ϵ for a convergence study where a second order method is used (i.e., p=2) and the mesh resolution is doubled (r = 2.) The value for the parameter F_S is based on Roache's experience of observing different computational studies. He finds that a value of three provides a reasonable error band for convergence studies of two grids. The GCI provides a suitable error band that is comparable to that used by experimentalists when they estimate experimental error as

two standard deviations. When $F_S = 3$ provides an error estimate that is considered too conservative the value can be lowered to a value of 1.25 if the order of convergence, i.e., p, can be verified using a convergence study that applies three grid resolutions or more. Richardson extrapolation provides a means for such an analysis. The following relation estimates the order of convergence for three mesh resolutions of any resolution ratio:

$$p = \frac{\ln\left(\frac{f_3 - f_2}{f_2 - f_1}\right)}{\ln(r)}. \tag{2.34}$$

Equation 2.34 provides a good measure of the actual order of convergence assuming that the mesh resolutions used are in the asymptotic range of convergence. This means that the mesh resolution is fine enough that error contributions from higher order terms in the Taylor series are negligible. The subscripts in Equation 2.34 denoted different mesh resolutions where the lowest numbers represent the finest grids. Another advantage to Equation 2.34 is that no error need be directly computed to get the order of convergence. All that is required are function values for different mesh resolutions.

2.9.2 A Method of Validation: Data Collaboration

For model validation, the purpose is to evaluate the performance of a model's calculations in comparison with a real physical system. This naturally leads to analysis that compares the model's predictions with experimental data. Before validation can begin, verification exercises must be executed to account for discretization error, iterative error, and code errors. Once these types of error are accounted for, the only remaining error to be addressed is error arising from modeling assumptions and simplifications. The problem of model validation brings the experimentalist and the computationalist together in a type of analysis that can identify issues in both experimental technique and modeling that can be very beneficial to both fields. This can also lead to better understanding of physical systems and better predication of such systems. The problem of model validation is illustrated in Figure 2.10. The plot depicted is a contrived example of model/data comparison. The ″o″ symbols represent an experimental measure of some quantity of interest. The line represents a model's prediction for the same quantity under the same conditions.

One such method is called data collaboration, and it is described by Feeley et al. (2004). In their description the authors deal largely with identification of parameters in the methane combustion reaction mechanism, but the method of data collaboration is an important tool wherever a model needs to be validated with experiment. In answering the question of data/model agreement, the authors introduce the notion of data consistency. To understand consistency, certain terminology must be introduced. Data collaboration is based on the analysis of what the authors call a dataset. A dataset involves a collection of data from a pool of experiments. In their nomenclature each experimental is represented by the letter ε, and quantities associated with each experiment are given the subscript e. The experimentalist is interested in measuring some quantity which is represented by Y_e. The measured value of that quantity is denoted y_e. Ideally, each experimental measurement includes a measure or estimate of experimental uncertainty in terms of an upper bound, u_e, and a lower bound, l_e such that:

$$l_e \leq (Y_e - y_e) \leq u_e. \tag{2.35}$$

Additionally, the computationalist applies a mathematical model to give a prediction of that quantity, Y_e. The model prediction is denoted as $y_m(\mathbf{x})$, where y_m is the model's prediction of the quantity of interest as a function of a number of model parameters, $\mathbf{x}$, which are typically adjusted to reflect the conditions of the experiment and which typically include uncertainty as well. For particular experimental measurement, y_e, u_e, l_e, and y_m form what is called a dataset unit. A collection of dataset units forms a dataset. As already stated, the model parameters will have upper and lower bounds to reflect the uncertainty in their values as well. This implies that a model can also give a range of output values as predictions for the quantity of interest since there is a range of possible values from its parameters. The authors assign each parameter, x_j, a symbol for their upper bound, β_j, and a symbol for their lower bound, α_j. The authors term the entire range of possible values for the model's parameter as a hypercube, $\mathcal{H}$. With all these definitions in place a definition of dataset consistency can be offered. The definition rests on two constraints:

$$\alpha_j \leq x_j \leq \beta_j \; (for \; j = 1, 2, 3, ...), \tag{2.36}$$

and

$$l_e \leq (y_m(\mathbf{x}) - y_e) \leq u_e. \tag{2.37}$$

In these expression, j denotes the set of input parameters to the model. For a given dataset, if there is no point x in the hypercube, $\mathcal{H}$ defined in Equation 2.36 that satisfies the constraints of Equation 2.37, then the dataset is said to be inconsistent. Otherwise, the dataset is said to be consistent. Consistency is a much more rigorous and quantifiable means of determining data/model agreement than visual inspection of a comparison plot.

The consistency analysis described by Feeley et al. (2004) can do more than determine consistency of a dataset. It can also identify which members of a dataset have the greatest effect on the consistency. Such analysis can lead to the identification of outlier data in the experimental set, or it can identify potential errors in the model itself. Another matter of importance in consistency analysis is the identification of what the authors call the primary variables. The primary variables are the model parameters that have the greatest impact on the consistency of a given dataset. The authors note that, in practice, even in models consisting of hundreds of parameters only a small subset of these parameters have a noticeable effect on the consistency. Determination of these active variables can be both the objective and a consequence of the consistency analysis. To understand how consistency analysis can yield this information more must be stated about how the authors measure consistency. The ease with which a dataset is found to be consistent can be viewed as a function of the width of the uncertainty bars with each experimental observation, y_e. If the experimental uncertainty is treated as variable the constraint shown in Equation 2.37 becomes:

$$l_e(1-\gamma) \leq (y_m(\mathbf{x}) - y_e) \leq u_e(1-\gamma). \tag{2.38}$$

Here, γ is a flexibility factor that when positive shrinks the uncertainty and when negative expands the uncertainty in the experimental measurement. In the consistency analysis the uncertainty width is adjusted over all the dataset units until the threshold value

of γ is found. The threshold value is the maximum value that γ can have where the dataset of interest is yet consistent. This threshold value is given another symbol, C_{ε}, and it is called the consistency measure. If this quantity is positive it means that the experimental uncertainties in the dataset had to be constricted to reach the threshold value and the dataset is consistent. If the consistency measure is negative, this means that the experimental uncertainties in the dataset had to be expanded to find the threshold value indicating an inconsistent dataset. It has already been shown that the values of u_e and l_e can influence the consistency of a dataset. If the range of model parameters were also adjusted, this would also affect the consistency. Another way of saying this is that the value of the consistency measure is affected by l_e, u_e, α_e, and β_e from each dataset unit. A perturbation in any of these parameters could conceivably affect the value of the consistency measure. An expression that illustrates this principle is formulated by the authors and is given here:

$$\Delta C_{\varepsilon} \leq \sum_{j=1}^{n} \left({\lambda_j}^{(\alpha)} \Delta \alpha_j + {\lambda_j}^{(\beta)} \Delta \beta_j \right) + \sum_{e=1}^{E} \left({\lambda_e}^{(l)} \Delta l_e + {\lambda_e}^{(u)} \Delta u_e \right). \tag{2.39}$$

In this relation, the symbol Δ denotes a perturbed quantity. Equation 2.39 basically states that the disturbance in any of the uncertainty bounds, whether they be experimental bounds or model parameter bounds, will disturb the consistency measure to some degree. The degree to which each property can disturb the consistency measure is expressed in the magnitude of each λ, called the Lagrangian multiplier for that property. The Lagrangian multiplier is a measure of the sensitivity of the consistency of the dataset to a particular uncertainty bound. These sensitivity measures can provide valuable insight into the behavior of a particular dataset. If a dataset is found to be inconsistent the sensitivities can indicate the most likely cause. For example, if the analysis of an inconsistent dataset shows that only a few Lagrangian multipliers are orders of magnitude larger that the others and these multipliers are associated with the bounds of an experimental measurement, then it is likely that that data point is an outlier and either needs to be removed from the dataset or the experimental error for that measurement needs to be re-evaluated. The same kind of analysis would apply to the bounds of a model parameter. If there are no

clearly dominant Lagrangian multipliers among the experimental uncertainties with an inconsistent dataset, then the model itself may need to be re-examined. Depending on the results of the consistency analysis errors can be identified on both the experimental and computational side of the collaboration.

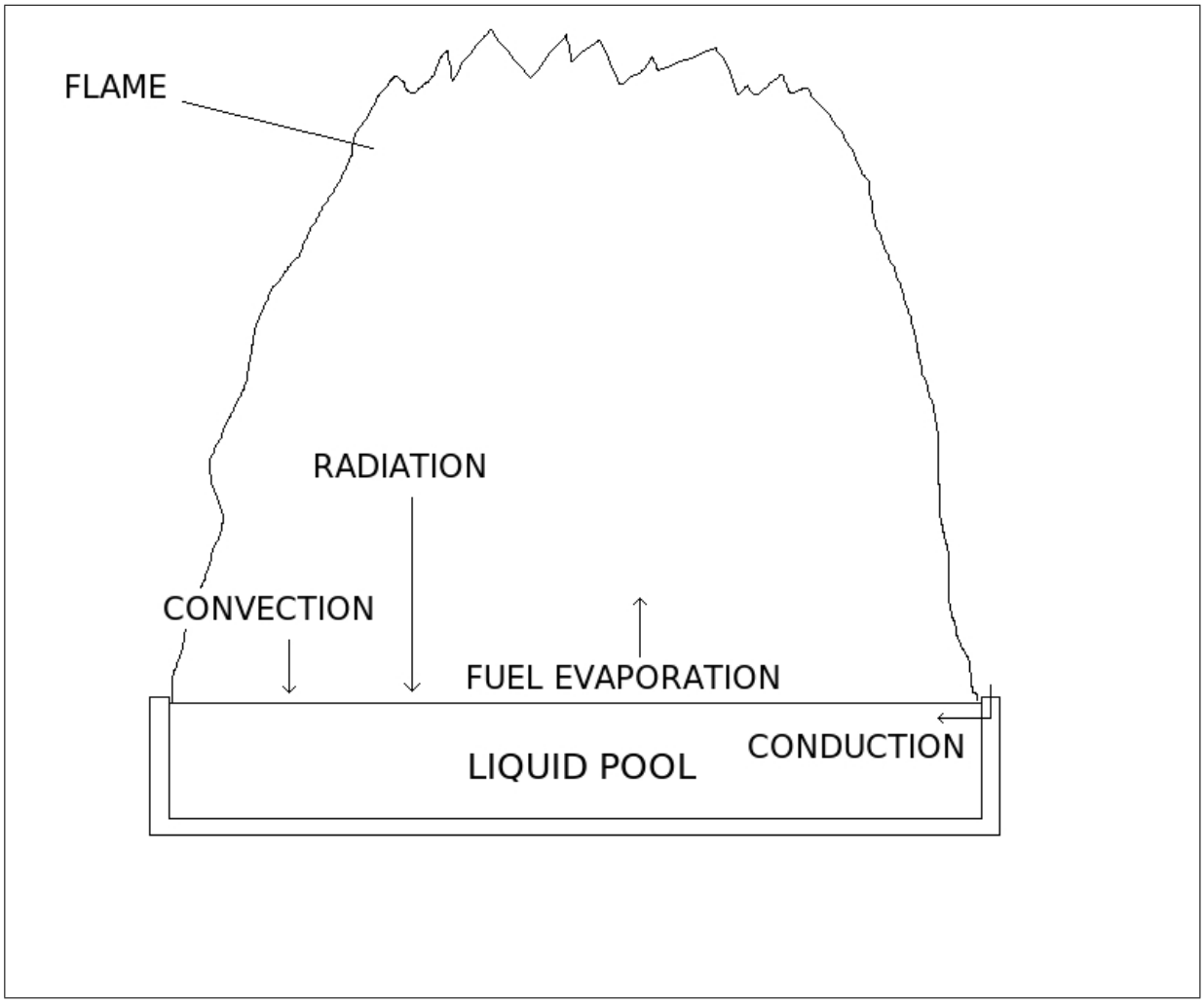

Figure 2.1: A pool fire with the modes of heat transfer labeled.

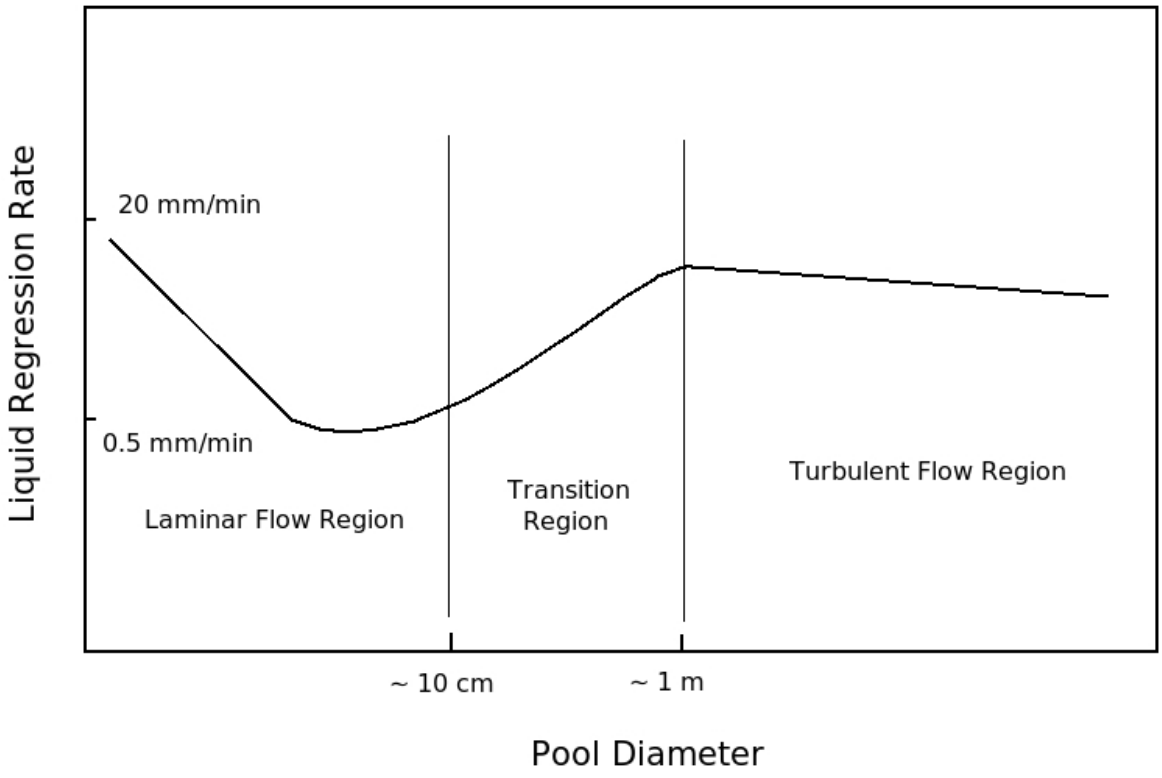

Figure 2.2: The qualitative behavior of liquid pool regression rate versus pool diameter based on Hottel's (1958) analysis.

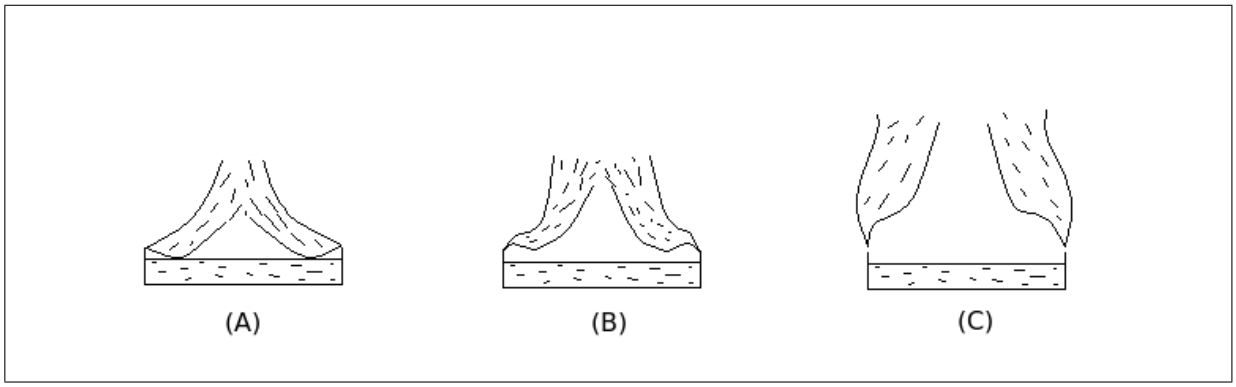

Figure 2.3: A sketch of basic pool fire flame shapes at different burning phases.

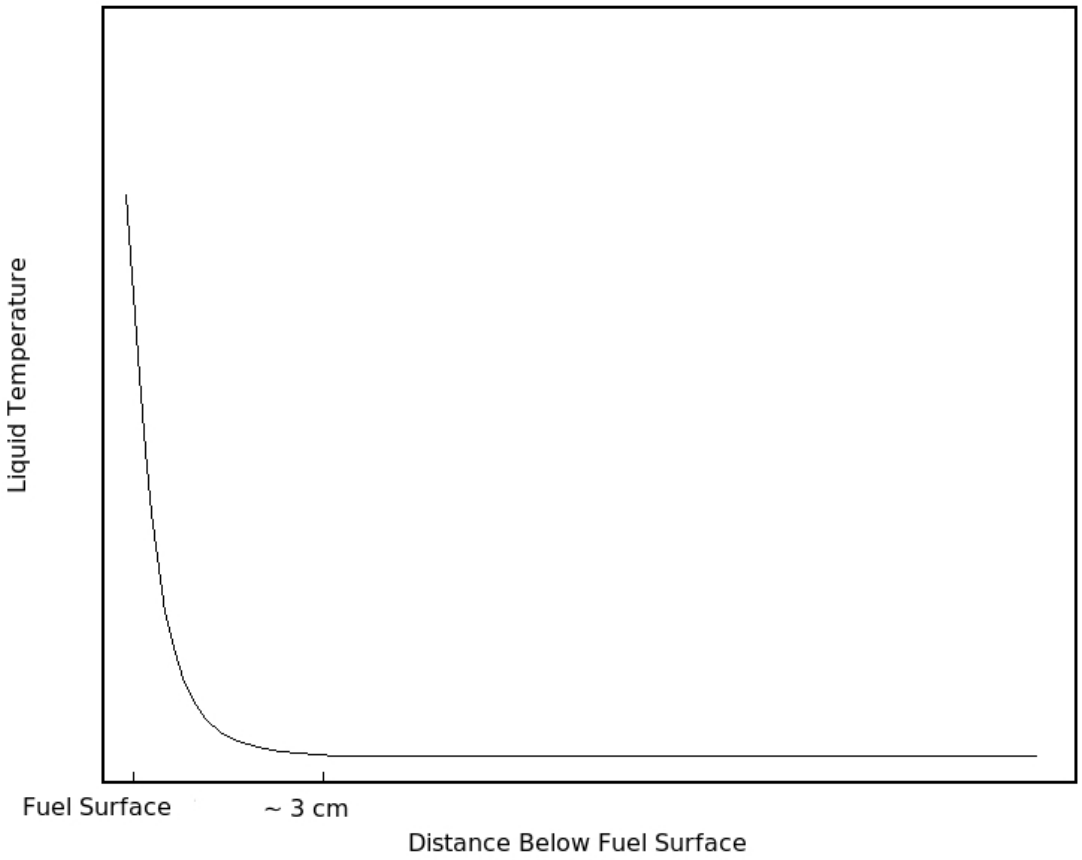

Figure 2.4: A qualitative sketch of temperature data common to most burning pools.

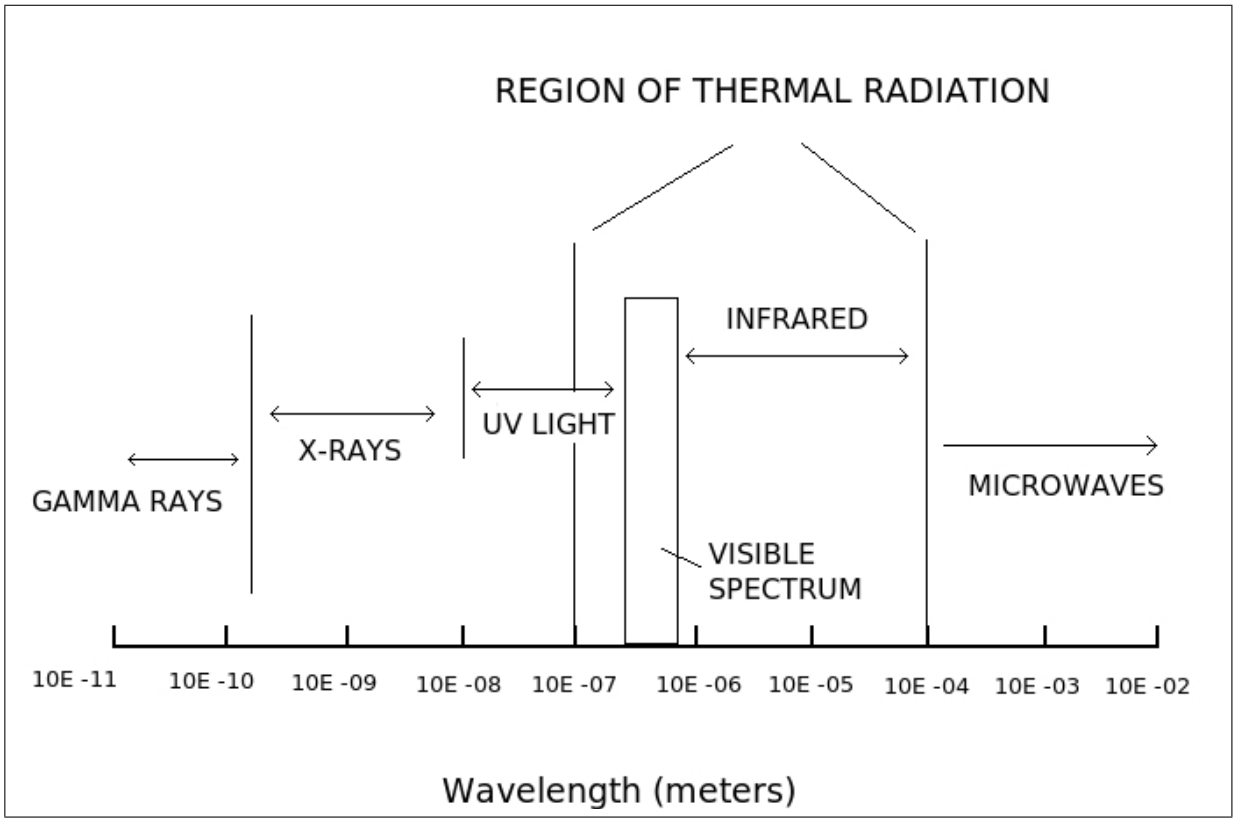

Figure 2.5: The electro-magnetic spectrum with the thermal radiation portion labeled.

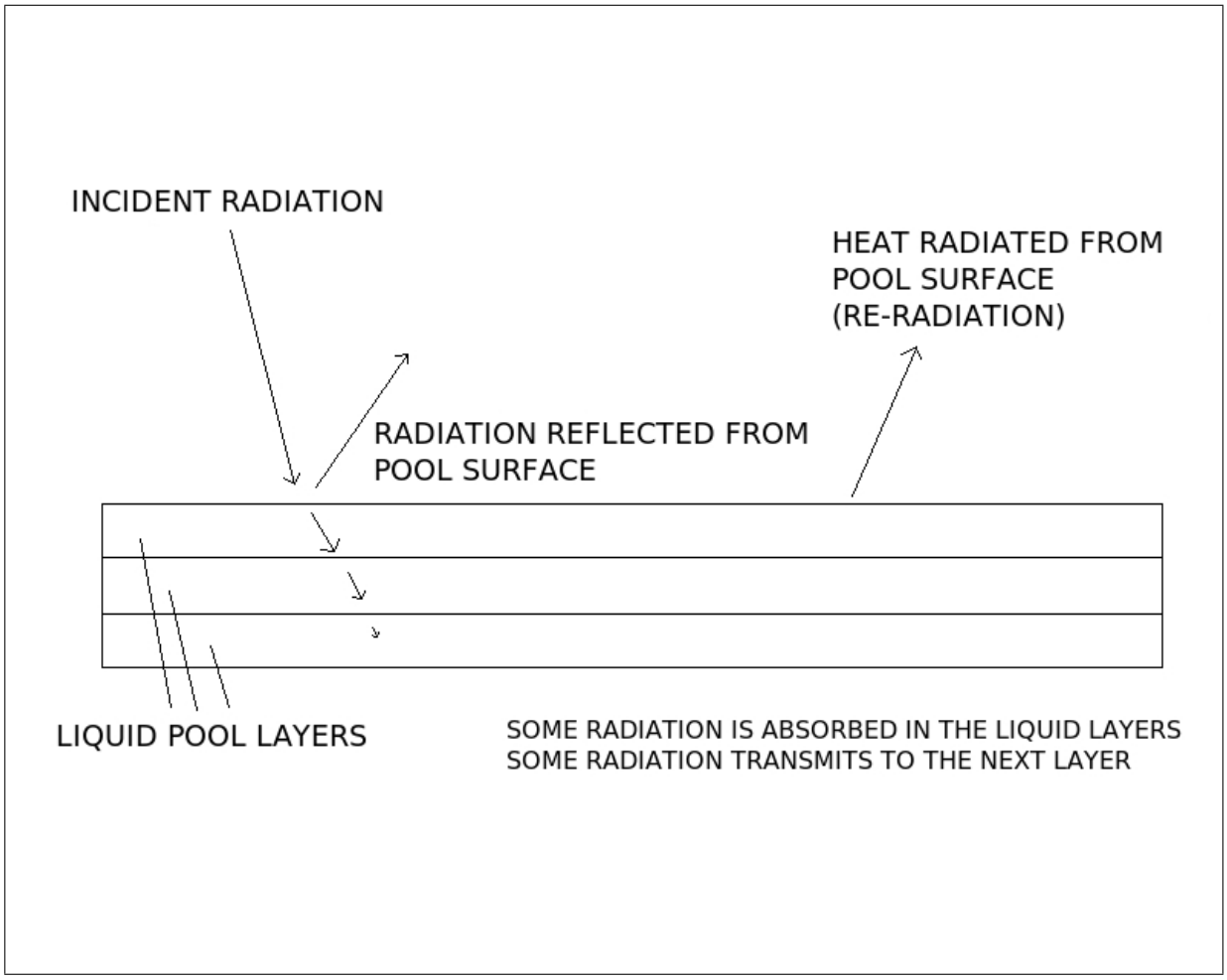

Figure 2.6: The dynamics of radiation interaction with the pool.

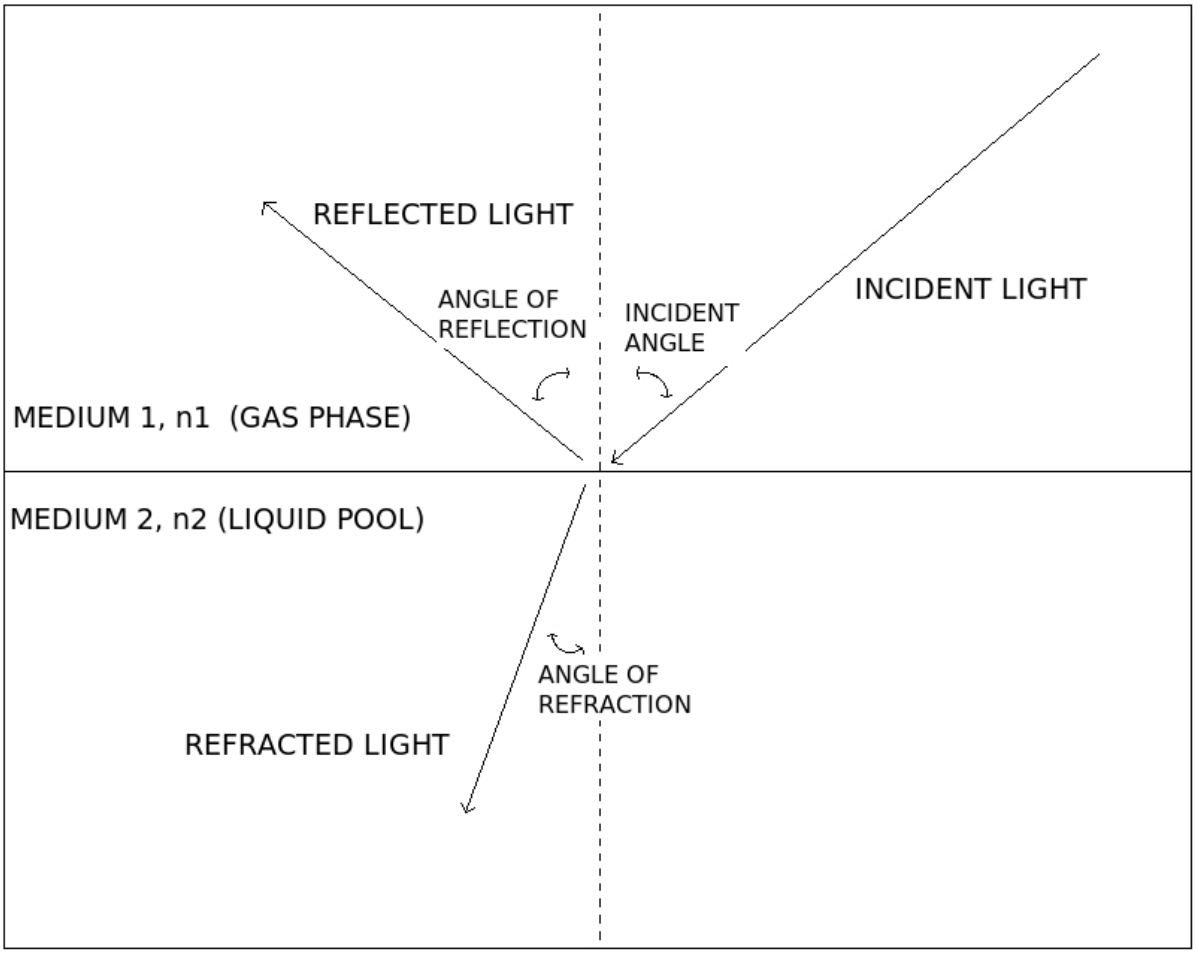

Figure 2.7: The reflection and refraction of light passing between two transparent media. Relevant angles are also shown.

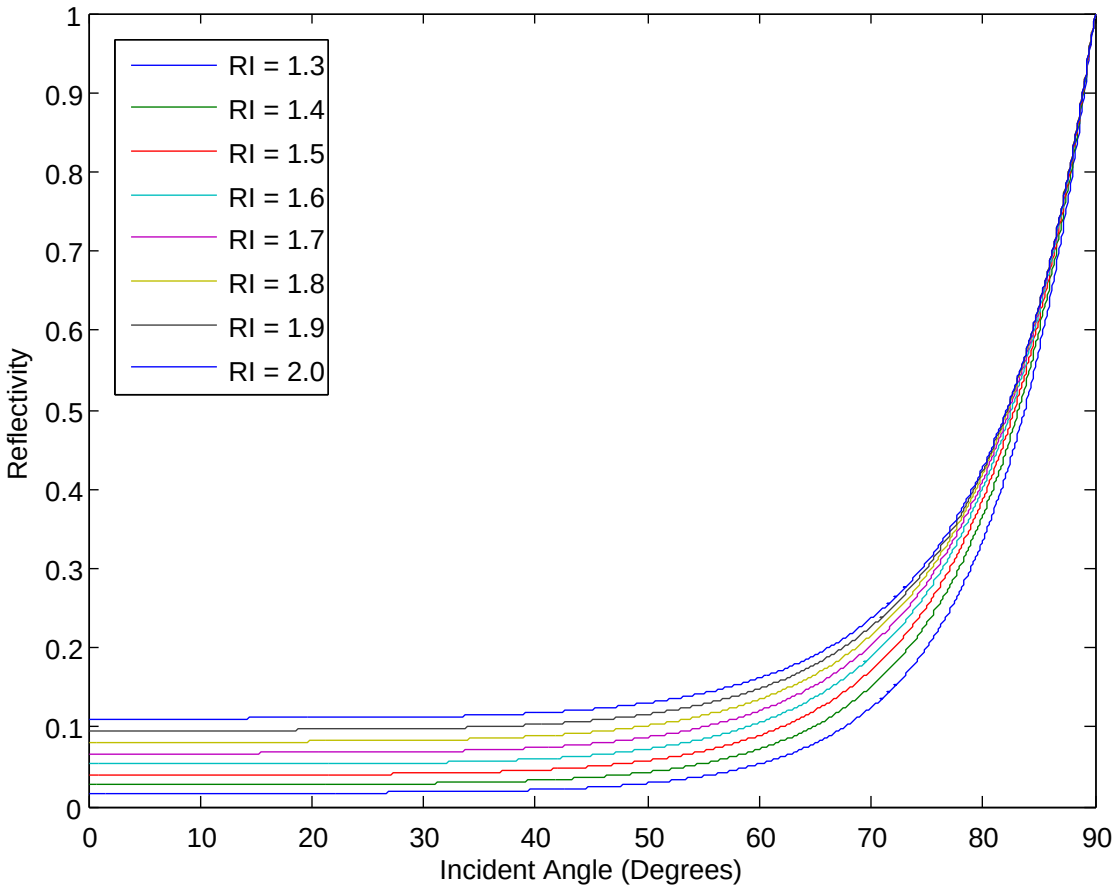

Figure 2.8: The surface reflectivities of incident radiation as a function of incident angle for media of different refractive indexes (RI). Incident angle is defined in Figure 2.7.

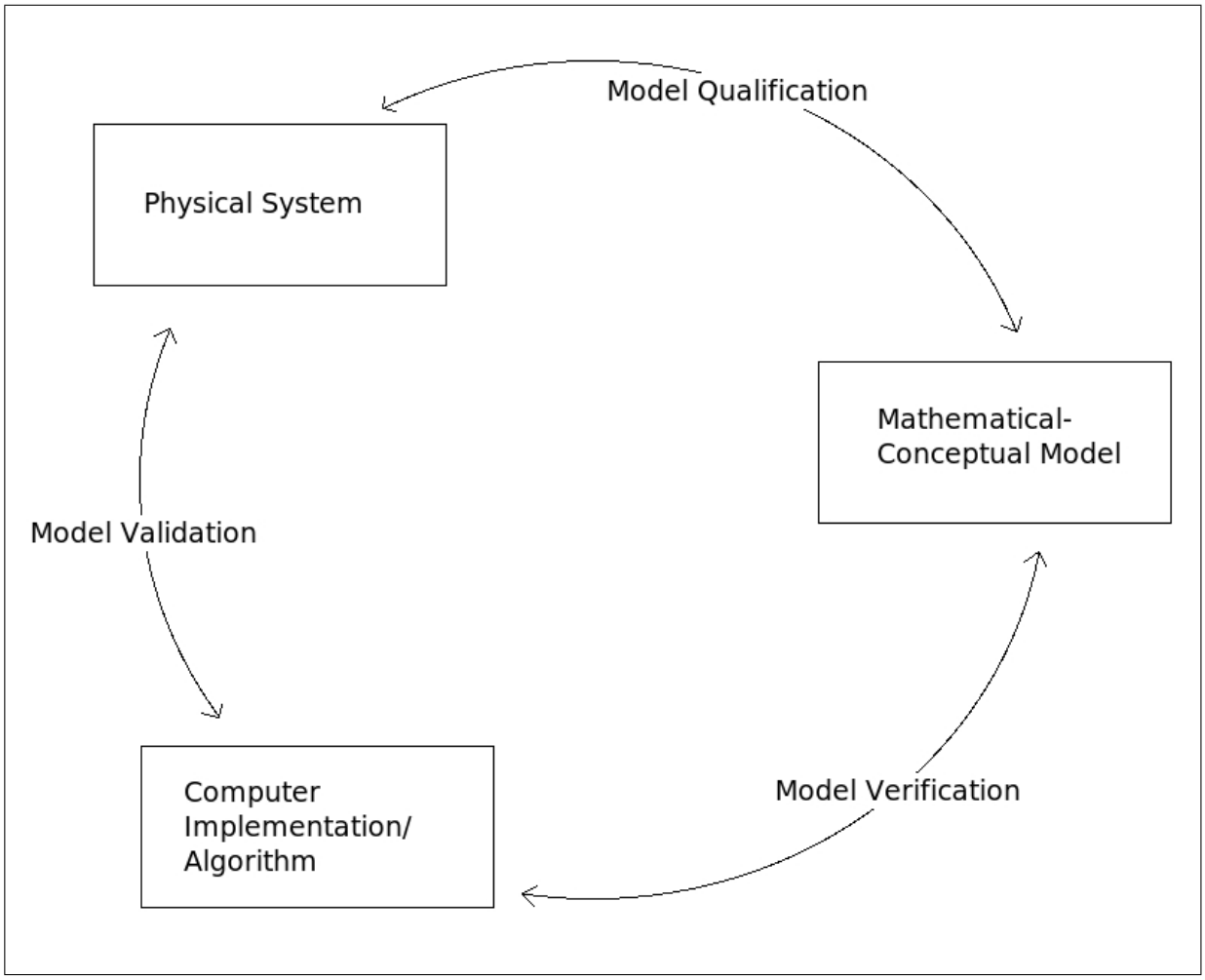

Figure 2.9: The process of computational model development adapted from a chart provided by Oberkampf and Trucano (2002).

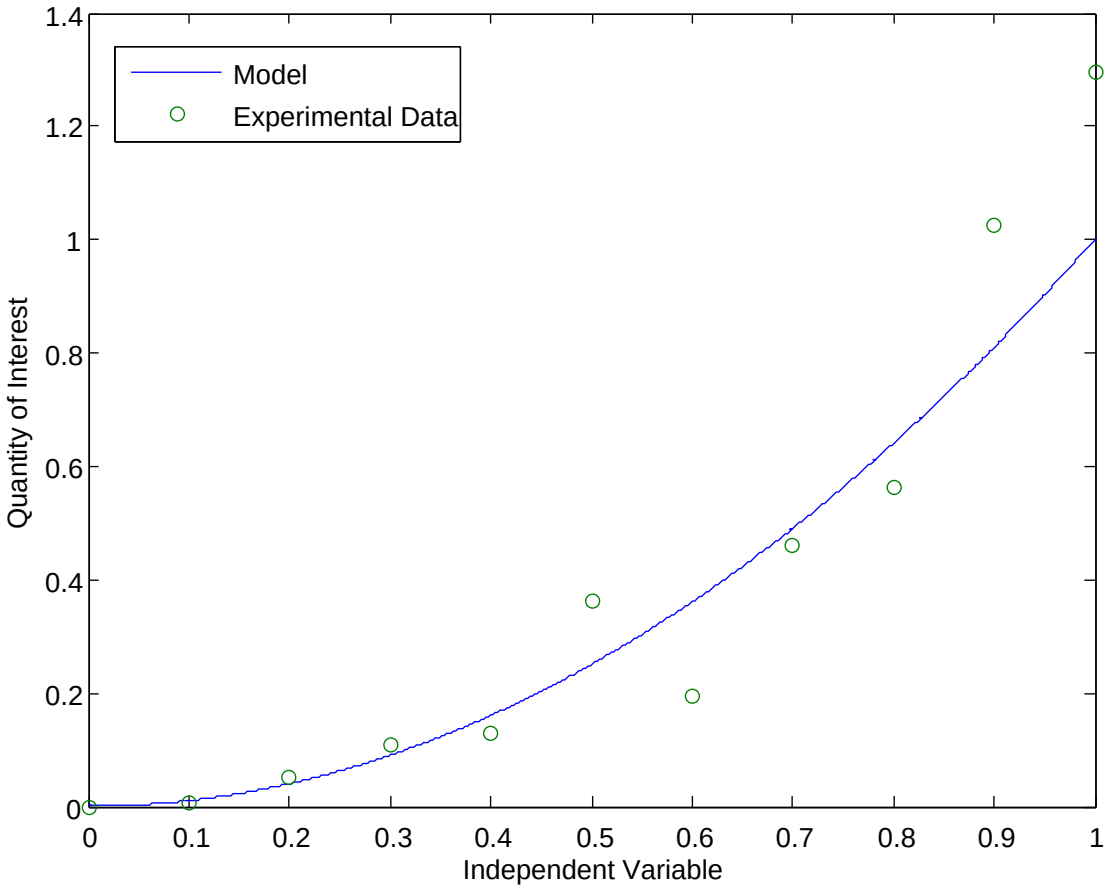

Figure 2.10: A contrived example of data/model comparison.

CHAPTER 3

RESPONSIVE BOUNDARY MODEL DESCRIPTION

3.1 Introduction

In formulating the model for a burning liquid pool it is important to start with the physical phenomena that are most critical to the pool's behavior. The previous chapter discussed these phenomena. Phenomena such as radiative heat transfer, the radiative properties of liquid fuels, and the energy balance to the pool surface are all important in determining the burning rate of the pool. Of course, not all physics are equally important to mass burn rate. For example, it has already been suggested in the previous chapter that horizontal conduction of heat across the pool could be neglected. To attempt to exhaustively describe the behavior of a liquid pool as it burns would be foolhardy as it would not be computationally tractable to model so much detail. The approach in designing the model for the Responsive Boundary is to keep it as simple as possible and yet still capture enough of the physical phenomena to be able to provide useful predictions of the mass burn rate of the pool as the fire evolves. The Responsive Boundary model is intended to be ported to the ARCHES fire simulation code and must provide accurate boundary data without adding significant computational burden to the fire code.

The Responsive Boundary model is written in C++, and it consists of approximately 4000 lines of code. It did not start this way, of course. The first steps in the development of the model involved a simple one-dimensional transient conduction model of the liquid. Heat could transfer through the layers of the liquid with the depth, but no conduction occurred horizontally. The liquid domain did not evaporate, and therefore did not change height with time. The liquid surface was exposed to a heat source and the temperature profiles were calculated. Eventually the model was expanded to include models that estimated the mass burn rate of the liquid at the surface, and a method was developed to simulate the

shrinking liquid domain as it evaporated. As the model took shape, the fuel was assumed to be only a pure, single-component. n-Dodecane was used to simulate the behavior of jet fuel (JP-8). The initial objective was to achieve the same average burn rate given the average thermal radiation input from the experiments performed by Blanchatt et al. (2006). For low wind cases the simple model predicted very accurate mass burn rates. For the higher wind cases the simple model over predicted the mass burn rate by about 6%. These initial results are encouraging, but not complete. The reflection of radiation for these early tests was set at 15%, whereas the literature can only justify a reflection of 10% at the highest. This likely led to an under-prediction of the burn rate. Additionally, one property that was not measured or known was the concentration of fuel just over the pool surface. Until the boundary model was coupled with ARCHES, this quantity could not be predicted. The simple model assumed that the fuel burned immediately and would not be present over the pool surface. In reality, there would be a vapor zone above the pool and its presence would inhibit the burn rate to some degree. Assuming no fuel above the pool would lead to an over-prediction of the mass burn rate. The initial results were encouraging nonetheless. Eventually the Responsive Boundary Model was expanded to include multicomponent fuels and the effects associated with multicomponent mass transfer. Finally, code was added to allow the Responsive Boundary model to be coupled with ARCHES.

3.2 Physical Domain and Conversation Equations

The domain of interest for the Responsive Boundary model is a volume of liquid fuel that is spatially divided into layers. The domain is one-dimensional, that is the temperature and other physical properties are allowed to change with the depth of the fluid. Figure 3.1 illustrates the domain of interest.

Although the domain depicted in Figure 3.1 shows a circular cross-sectional area, the shape of the pool can take any form as long as it is constant with depth. As mentioned already, the domain is divided into nodes that represent the properties of the fluid at each point along the depth of the pool. The top node represents, at least partially, the properties of the pool surface. The pool surface temperature is actually extrapolated from the top three nodes of the pool domain, which will be described in more detail later. Heat input

to the pool through the pool surface is accounted for in the top nodes of the pool.

3.2.1 The Energy Balance

The first fundamental equation in the Responsive Boundary Model is the conversation of the energy in the pool. It is assumed that there is negligible heat loss through the sides and through the bottom. Since most of the heated liquid is within about 3 centimeters of the surface as long as the pool is deeper than 3 centimeters this is a good assumption for the bottom of the pool and the bulk of the sides as well. Since the domain is one-dimensional, the energy balance is a continuum model that is also one-dimensional in space (denoted by the spatial coordinate z) and transient. The continuum energy balance is expressed as:

$$\frac{\partial E}{\partial t} = \rho V C_P \frac{\partial T}{\partial t} = kV \frac{\partial^2 T}{\partial z^2} + Aq^{''}_{radiation} + Aq^{''}_{convection}. \tag{3.1}$$

In this relation, E is the energy of the system, T is the liquid temperature which is a function of both time and liquid depth, ρ is the liquid density, V is the system volume, C_P is the liquid heat capacity, k is the liquid thermal conductivity, A is the surface area of the pool, $q^{''}$ represents heat fluxes to the pool due to thermal radiation and convection. Note that there is no conduction term because for larger pools, the intended system for this model, heat conduction is negligible compared to the other modes of heat transfer. By isolating the partial derivative term with respect to time the following form of the energy balance results:

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial z^2} + \frac{q^{''}_{radiation}}{\rho C_P \Delta z} + \frac{q^{''}_{convection}}{\rho C_P \Delta z}. \tag{3.2}$$

Here, α is the thermal diffusivity of the liquid, and Δz is the length of the node. Note that the division performed to acquire Equation 3.2 has eliminated both volume and area from the energy balance which demonstrates again that the shape of the cross section of the pool is unimportant as long as its form remains fixed throughout the depth of the pool. The heat fluxes due to radiation and convection are treated as source terms in the discretization of the energy equation and do not have the same value from node to node. Initially, it was decided to place these terms only at the surface node, since these are surface phenomena. However,

this led to a computational problem. With this implementation, the surface temperature, or the temperature of the top node and the overall temperature profile changed with the number of nodes used in the simulation. The more nodes used the smaller the value of the spatial step size, Δz, and as the step size decreased the surface temperature would increase without bound. Ideally, as the step size decreases the temperatures should arrive asymptotically to a certain value. The solution to this problem came with the realization that radiation absorption is not really a surface phenomenon since radiation penetrates somewhat into the depth of the pool. If the length of the top node represents the depth of radiation penetration then the temperature at the top node would be representative of the real temperature. Since node sizes change with the number of nodes and it is desired to be able to change the number of nodes for convergence test purposes assuming that only the top node absorbs radiation is not acceptable. Instead, it is assumed that thermal radiation can be absorbed into a certain depth of of the liquid pool and the nodes in that penetration zone would absorb certain amounts of the radiation input. Ideally, the Beer-Lambert law (Equation 2.7) should be applied to determine the penetration depth and the amount of radiation absorbed at each node in the penetration zone. However, the values of the extinction coefficient for various fuels at various wavelengths of light are not always available. It is therefore simpler to assume that the most hydrocarbon fuels have similar radiation penetration depths and that an exponentially decaying function of the same form as Equation 2.7 can describe the intensity of thermal radiation with depth. Examination of temperature profile data in burning liquid hydrocarbon pools suggests that the temperature drops to nominal value by about 3 centimeters of depth as shown in Figure 2.4. This is also consistent with the observations made in the literature that the bulk of the thermal radiation is absorbed within about 1 centimeter of the surface of the pool. The coefficient for the decaying exponential model is chosen such that about 97% of the thermal radiation is absorbed within 3 centimeters of the pool's surface.

With the decaying exponential model the code can determine how many nodes lie within the 3-centimeter radiation penetration zone, and it can calculate what percentage of the thermal radiation is absorbed in each node in that zone. With this implementation in place, simulations can be run with various numbers of nodes without producing unbounded

temperature values as the nodes increase. The temperature profile tends to asymptotic values as the spatial step decreases.

Observation of Equations 3.1 and 3.2 reveals that there is no term for heat loss due to evaporation of fuel. There is a term for this energy loss and it is expressed:

$$q''_{evap} = m''_{evap} \left[\Delta H_{evap}(T_s) + \int_{T_o}^{T_s} C_P(T_s)\, dT \right]. \tag{3.3}$$

In this expression the heat of vaporization and the heat capacity of the liquid are functions of the surface temperature of the liquid T_s. The energy loss to the pool system from evaporation includes two terms in Equation 3.3. The first term represents the heat needed to vaporize the liquid, and the second term represents the energy used to heat the liquid to its temperature at the time of evaporation. The sensible heat is calculated from a standard temperature, T_o which is chosen as the temperature of the surrounding environment and the initial liquid temperature. The energy loss due to evaporation was initially used as a separate term in Equation 3.2, but this created another problem. The heat loss caused the surface temperature to drop below the temperature of the liquid layers just below the surface. In the physical system the cooled liquid would be more dense and would simply recirculate and mix with the warmer liquid layers below. Unfortunately, the pool model does not include variable liquid density and cannot include this effect. One option to resolve this issue is to include a momentum balance that can account for density differences and the resulting mixing effects. Such an addition would increase the complexity of the Responsive Boundary model and require a higher dimensional model for the liquid system. In place of this, the energy loss due to evaporation is simply subtracted from the absorbed radiation input, and the mixing due to density difference is assumed to occur quickly so that it would not need to be modeled directly. While this technique is not as physically realistic, it maintains the important terms of the energy balance and maintains model simplicity. The error inflicted by such an assumption can be judged from the validation process.

A similar approach was also used with the convective flux. Any energy flux applied only at the surface node causes sensitivity problems with variation in the spatial step size

as was seen from the issue with the radiative heat flux. Instead of applying the convective flux directly to the energy balance at the top node, the energy was added to the absorbed radiation input, which, as has already been mentioned, is distributed through the top 3 centimeters of the pool. Again, this is not technically physically realistic, but it can be justified by the fact the thermal contribution from convective heat flux is generally much less than that from radiative heat flux, and applying the convection heat flux in this manner allows a more stable and consistent performance from the model. Also, since it is established that their is a degree of recirculation in the top layers of the pool, it can be assumed the the convective heat flux absorbed at the pool's surface is quickly distributed throughout the top layers just as the radiative heat flux. Error inflicted from this technique can likewise be examined in the validation process.

3.2.2 Mass and Species Balances

The other important continuum equation that forms the basis of the Responsive Boundary model is the mass balance in the form of a liquid height balance:

$$\frac{dH}{dt} = \frac{-m^{''}evap}{\rho}. \tag{3.4}$$

In this equation, H is the liquid height. This mass balance assumes a non-reactive system that loses mass only by means of evaporation. Note that the size of the domain and therefore the size of the spatial step size used in the energy balance are dictated by the value for the liquid height and will change as liquid height changes. The method for addressing this problem will be discussed later in this chapter.

The last continuum model used in the Responsive Boundary model is the species balance, and it only applies for use with multicomponent fuels:

$$\frac{dC_i}{dt} = \frac{-N^{''}i, evap}{H}. \tag{3.5}$$

In this equation, the evaporation flux ($N^{''}i, evap$) is expressed on a molar basis. C_i is the molar concentration of the i^{th} species. Initially, a diffusive flux term was included to account for differences in species concentration with the depth of the liquid, but the

calculation of the diffusive fluxes were somewhat computationally expensive and the fluxes themselves were found to be negligible. A simpler approach is to assume uniformity throughout the liquid due to agitation of the liquid during the burning process and assume that the only way a species leaves the pool is through evaporation from the surface. Mixing in the top layers of the pool is a reasonable assumption given the discussion on the recirculation due to the cooling of the surface. According to Blinov and Khudyakov (1957), in some cases mixing throughout the pool's depths is justified where the density differences between the components is sufficiently large.

3.2.3 Discretization Scheme

To make the solution of the continuum equations possible it is necessary to use some sort of discretization method in both space and time. All three continuum equations are transient and require a time stepping algorithm. The chosen method for time stepping is the fourth order Runge-Kutta method. Constantinides and Mostoufi (1999) describe the method in their text. Whether the variable is liquid temperature, height, or species concentration the Runga-Kutta method estimates the value of the quantity of interest as:

$$U_i + 1 \approx U_i \frac{\Delta t}{6} \left(k_1 + 2k_2 + 2k_3 + k_4\right). \tag{3.6}$$

In this formulation, U represents the quantity to be solved from the continuum equation (temperature, liquid height, or species concentration.) The subscript i along with U indicates the quantity at different time steps, and Δt is the time step size. The different $k-$parameters are estimates of the time derivative and are defined:

$$k_1 = f(t_i, U_i), \tag{3.7a}$$
$$k_2 = f(t_i + \frac{\Delta t}{2}, U_i + \frac{k_1}{2}), \tag{3.7b}$$
$$k_3 = f(t_i + \frac{\Delta t}{2}, U_i + \frac{k_2}{2}), \tag{3.7c}$$
$$k_4 = f(t_i + \Delta t, U_i + k_3). \tag{3.7d}$$

The functions f described in Equation 3.7 are the values of the temporal derivative according to the respective differential equations. In the case of the Responsive Boundary

model, they are the right-hand sides of Equations 3.2, 3.4, and 3.5. The Runga-Kutta method described in Equation 3.6 does not give exact solutions to the continuum equations at each time step. The method can be derived from truncated Taylor series, and the neglected terms of the series represent the error in the method. The method is termed fourth order because the largest term in the error contains a dependence on the time step raised to the fourth power. It is common to say that such a method gives fourth order error with respect to time step:

$$[Error]_{RK4} \approx O\left(\Delta t^4\right) = k\Delta t^4. \tag{3.8}$$

The energy equation (Equation 3.2) contains a spatial derivative, and this too must be discretized in order to be solved. It is important to remember that the spatial derivative term describes the difference in diffusive heat fluxes in and out of each node in the pool domain. For the spatial nodes in the pool that are not at the top or the bottom of the pool the following approximation is used:

$$\begin{aligned} \frac{\partial^2 T}{\partial z^2} &= k(z_i)\left(\frac{\frac{T_i - 1 - T_i}{\Delta z} - \frac{T_i - T_i + 1}{\Delta z}}{\Delta z}\right) + O(\Delta z^2) \\ &= k(z_i)\frac{T_i - 1 - 2T_i + T_i + 1}{\Delta z^2} + O(\Delta z^2). \end{aligned} \tag{3.9}$$

Here, T_i is the temperature at node i, $T_i - 1$ is the liquid temperature at the node above node i, and $T_i + 1$ is the liquid temperature at the node below node i. The symbol k represents the liquid thermal conductivity at node i, and it is a quantity that can vary with location. The middle section of Equation 3.9 demonstrates that this term is the difference between the flux at the upper boundary of the node and the lower boundary of the node. The far right-hand side of the same equation shows that this reduces to the common, central difference, second order discretization for a second derivative term. The derivative terms for the nodes at the top and bottom will use the same formulation as shown in Equation 3.9 except that the flux in one of the boundaries will be zero. This reflects the assumption that there is no diffusive heat flux through the top and bottom of

the pool domain. In summary, the discretization of the continuum equation, if properly executed, should give discretization error that is second order with respect to spatial step size and fourth order with respect to temporal step size. One major task of verification is to confirm this order of accuracy in the model's results.

3.3 Mass Burn Rate Model

The overall objective for the Responsive Boundary model is to predict the rate at which the fuel in a pool fire vaporizes. This is the primary output of the model. From thermodynamics it is known that a liquid species below its boiling point will vaporize into open air as long as the vapor concentration of the species in the air is below its saturation point. The rate at which it vaporizes depends on the vapor pressure of the species and the liquid temperature at the liquid/gas interface. Below the boiling point, the vaporization of the liquid is mass transfer controlled. This means that the rate of vaporization will be controlled by the diffusion of the species from the surface and as well as the convective conditions in the surrounding fluid. For example, if there is no crosswind over the surface of the pool, then the vaporization will be driven by buoyant or free convection which is driven by density differences in the gas phase. If a crosswind is present then forced convection will control the rate of mass transfer depending on the wind speed. Both phenomena can be present. Forced convection generally produces higher mass transfer rates than those produced from buoyant convection.

If the liquid reaches its boiling point its rate of vaporization is no longer controlled by mass transfer limitations. Instead, the liquid vaporizes as fast as energy is supplied to it. Since it is the common observation that for all pure component fuels and many multicomponent fuels the surface temperature does not reach the boiling point of the liquid, most effort is placed on modeling the mass transfer controlled regime. To estimate the mass transfer rate the Responsive Boundary model uses film theory for mass transfer between a fluid/fluid interface. Figure 3.2 illustrates this theory.

Film theory states that all resistance to mass transport occurs in a thin film between the two fluid phases. In the case of a pool fire this is assumed to be a thin gas film just above the surface of the pool. All concentration gradients occur within this film. In Figure

3.2, d is used to represent the thickness of the film, C_i is the concentration of the fuel at the bottom of the film and is assumed to be the saturation concentration of the fuel at the liquid surface temperature. C_{bulk} is the bulk concentration of the fuel in the bulk gas phase. According to film theory as described by Seader and Henley (1998) the molar flux is calculated from this model:

$$N^{''}evap = C_t k_c \ln\left(\frac{1 - x_{bulk}}{1 - x_i}\right). \tag{3.10}$$

Conversion to mass flux is a simple calculation involving the molar mass of the fuel. Equation 3.10 applies to single component mass transfer through a stagnant film. C_t is the film's molar concentration and can be calculated from the ideal gas law for the pool fire system. The symbol k_c is the mass transfer coefficient, and x_i and x_{bulk} are the mole fractions of the fuel at the bottom of the film and at the bulk gas phase, respectively. This form of the mass transfer equation may be applied when bulk mass flow is significant, and it is in the case of a pool fire.

The estimation of the mass transfer coefficient relies on heat transfer coefficient correlations, and depends on the mode of convection over the surface of the pool. For buoyant convection, the correlation used is appropriate for transfer from the top of a flat, horizontal plate. This model is described by Incropera and Dewitt (2002). Additionally, since heat and mass transfer are occurring actively at the surface of the pool, a model for the Grashof number described by Bird et al. (2002) is used in conjunction with the correlation. The Grashof number describes the effect of buoyancy on heat and mass transport. There are two types of Grashof number. The first, simply called the Grashof number or the thermal Grashof number, describes buoyancy force due to density differences caused by difference in temperature. The other is called the diffusional Grashof number, and it describes buoyancy force due to density differences caused by differences in species concentration. They are both defined here:

$$Gr = \frac{g\beta\left(T_1 - T_0\right)L^3}{\nu^2}, \tag{3.11}$$

and

$$Gr_{\omega} = \frac{g\xi\left(\omega_1 - \omega_0\right)L^3}{\nu^2}. \tag{3.12}$$

Gr, is the symbol for the Grashof number. L is a characteristic length that depends on the geometry of the system. In the case of the pool, the characteristic length is the area of the pool divided by its perimeter. The symbol g represents the acceleration due to the earth's gravitational force, β is the thermal expansivity or the change in density of the film with temperature, ξ is the change in density with mass fraction, ω, and ν is the kinematic viscosity of the fluid of the film. Bird et al. (2002) state that when the thermal and compositional effects on density change are both important the Grashof number in the mass transfer correlation can be replaced with the sum of the thermal and diffusional Grashof numbers. This methods applies only if the Prandtl number, Pr and Schmidt number, Sc of the film have similar values. Taking all the above into consideration, the buoyant Sherwood number used in the Responsive Boundary model is:

$$Sh_{free} = 0.27\left(\left(Gr + Gr_{\omega}\right)Sc\right)^{\frac{1}{4}}. \tag{3.13}$$

In the case of forced convection, Incropera and Dewitt (2002) provide the heat transfer correlation that is utilized to get a Sherwood number for forced convection. This correlation is the average Sherwood number for forced convection over a flat plate:

$$Sh_{forced} = \left(0.037Re^{\frac{4}{5}} - 871\right)Sc^{\frac{1}{3}}. \tag{3.14}$$

The Reynolds number used in equation 3.14 is expressed by Equation 2.2. Incropera and DeWitt also describe a means to estimate the overall Sherwood number in cases were both buoyant and force convection effects are present:

$$Sh^3 = \left(Sh_{free}\right)^3 + \left(Sh_{forced}\right)^3. \tag{3.15}$$

The calculation of the mass transfer coefficient arises from a rearrangement of the definition of the Sherwood number:

$$k_C = \frac{ShD_{ab}}{L}. \tag{3.16}$$

In this equation, D_{ab} is the binary diffusivity of the fuel species with air, and L is the same characteristic length described for the Grashof numbers (Equations 3.11 and 3.12.)

For multicomponent interphase mass transfer, Taylor and Krishna (1993) provide the appropriate description for film theory with multicomponent systems. The system is assumed to be a stagnant layer of air above the pool through which each of n species can diffuse. This type of diffusion is called Stefan diffusion, and this assumption is necessary to obtain each of the n species fluxes through the film. The species fluxes can be calculated from properties on either side of the film:

$$(\mathbf{N}) = C_t[\beta_o][k_o][\Xi_o]\left(\mathbf{x_o} - \mathbf{x}_\delta\right) = C_t[\beta_\delta][k_\delta][\Xi_\delta]\left(\mathbf{x_o} - \mathbf{x}_\delta\right). \tag{3.17}$$

Here, $(\mathbf{N})$ represents the vector of molar fluxes for each volatile species, C_t is the total film gas concentration. Each quantity in brackets represents a matrix. The matrix represented by the symbol β is the bootstrap matrix which allows the computation of the diffusive flux for the n^{th} species. The form of the bootstrap matrix depends on the type of diffusion assumed for the calculation. The options described by Taylor and Krishna are equimolar counterdiffusion, Stefan diffusion, and fixed flux ratios. As already stated, Stefan diffusion is assumed for the evaporating pool system. The matrix represented by k is the mass transfer coefficient matrix and its composition depends on the mole fractions of species and the binary mass transfer coefficients computed from Equation 3.16. The matrix labeled by the symbol Ξ is the factor that accounts for the effects of bulk flow. This matrix depends on the values for $(\mathbf{N})$. This makes equation 3.17 implicit with $(\mathbf{N})$, and it requires an iterative solver to obtain the species fluxes. The Responsive Boundary code utilizes a fixed point type iteration for the cases where the liquid surface temperature is far from its bubble point. As the bubble point is approached, the fixed point algorithm fails to converge, and in this case a Newton-Raphson iterative method is used to obtain the solution. The definitions for the various matrices involved in Equation 3.17 can be found in the film theory chapter of the text authored by Taylor and Krishna. In Equation 3.17, x represents the mole fractions

of the various volatile species. The subscript o denotes properties computed on the side of the film near the pool, while the subscript δ denotes the same properties computed on the gas phase side of the film. The Responsive Boundary model uses the properties closest to the liquid pool.

In order to calculate the properties on the liquid side of the film it is necessary to use a vapor-liquid equilibrium (VLE) model. The Responsive Boundary model utilizes a simplified form of the modified form of Raoult's law described by Walas (1985). The model assumes that since the system is at atmospheric pressure (about 1 bar), the Poynting Factor and the ratio of liquid and gas phase fugacities will be near to unity. This gives a formula for the equilibrium gas phase composition:

$$y_i = \frac{\gamma_i P_{s,\,i}}{P} x_i. \tag{3.18}$$

Here, y_i and x_i are the gas phase and liquid phase mole fractions of species i, respectively. $P_{s,\,i}$ is the vapor pressure for species i as a function of the liquid surface temperature, and P is the system's fluid pressure. The parameter, γ_i is the liquid activity coefficient for species i. This parameter describes the thermodynamic deviation from ideal solution behavior due to mixing of unlike components, and it is a function of system temperature, pressure, and composition. For the Responsive Boundary model, this parameter is estimated using the Wilson equations as described by Walas (1985). The binary parameters used in the Wilson equations are estimated using a single-parameter correlation developed by Gothard et al. (1976). This correlation uses the difference between the solubility parameters of two pure species to correlate their interaction parameters for the Wilson equations. The method is developed and works best for mixtures of hydrocarbon liquids. The major limitation of the Wilson equations is that they cannot predict liquid phase splitting, which is satisfactory for the intended uses of the Responsive Boundary model. Besides the composition of the gas phase film, the activity coefficients from the Wilson equations are also used for the purpose of calculating bubble point temperatures for liquid mixtures.

3.4 Physical Property Estimation

Examination of the conservation continuum equations that form the basis of the Responsive Boundary, as well as the models for VLE and mass transfer show that these models incorporate certain physical properties of the fuel species involved. Given the number of possible fuel species that could be used by the Responsive Boundary model, there is not enough data available in the literature to specify all these properties using experimental data. Most of these properties are functions of the system temperature, pressure, and composition making the determination of these properties even more cumbersome. Some of these properties include liquid and gas phase density, thermal conductivity, heat capacity, binary diffusivity, viscosity, vapor pressure, etc. Instead of trying to gather data on all these properties as functions of system conditions for each conceivable fuel species, it is simply easier to use property estimation methods that are both theoretical and empirical and that rely on pure component properties that can be programmed into the model. Poling et al. (2001) gather numerous property estimation methods for each of these physical properties into one text, and most of the property estimation methods applied in the Responsive Boundary model are found there. The calculation of liquid density is one exception. At the beginning of the calculations, the liquid density is estimated based on the composition of the liquid using methods described by Walas (1985), and by Thomson et al. (1982). After the liquid density is calculated, the model assumes no change in the liquid density. The property estimation methods described by Poling et al. (2001) are accompanied by comparisons of their performance with measurements of the properties from experiment, and they include error. Each method in the Responsive Boundary model that involves property estimation is tested to be sure that the results are consistent with these comparisons in order to verify that the method is working as it should. Each method in the Responsive Boundary model involving property estimation has documentation to point to the specific method used and where its description can be found in Poling et al. (2001). The important point here is that each estimation method has its strengths and weaknesses, and that there is error in every calculation that must be taken into account during the model validation process.

3.5 Liquid Drop and Interpolation

A first glance at the fundamental continuum equations that form the basis for the Responsive Boundary model should give the reader the impression that it is a relatively simple one-dimensional, transient model. However, there is a phenomenon that complicates this otherwise simple model. The vaporization of fuel removes mass from the system causing the liquid level to drop. Since the computational domain is the liquid itself, the domain shrinks with time. This raises questions of how the domain should be spatially partitioned as it shrinks, where the temperature values will be tracked in the domain as it changes size, and how the approach applied will affect the discretization error in the calculations.

The domain is divided into a certain number of nodes as illustrated in Figure 3.1. The number of nodes used in the domain is fixed regardless of the size of the domain with time. Temperature values are calculated from the energy balance at the center of each node. With each time step new values are calculated for the temperatures at each node, and new values for the liquid composition and the liquid height are calculated. Once these values are determined the domain is redrawn as shown in Figure 3.3.

With each time step the liquid height drops slightly and a new domain is created with the new liquid height. The same number of nodes are used to divide up the new domain, but now the center of these nodes are now slightly offset in location when compared to the locations of the nodes in the old domain. Comparison of these new nodal positions with the old positions is used to obtain new temperature values via interpolation of the old temperature values. Also, because the size of the domain shrinks with time and the number of nodes remains constant, the spatial step size will shrink as well. This approach also corresponds well to the physical situation. When the model calculates the new surface temperature, that temperature does not actually exist because that top layer evaporates making the new surface temperature the temperature of the liquid just below the old surface. The drop in temperature due to the evaporation of the top layer is significant because of the sharp drop in liquid temperature with depth as illustrated in Figure 2.4. The heat input to the pool must constantly heat the top layers to a point and then heat the colder layers underneath as they emerge at the top due to evaporation. This physical process delays the heating process of the surface of the liquid and is critical for the correct

prediction of the mass burn rate.

Interpolation methods used to obtain new temperature values are approximate and provide another source of error. The Interpolation method used with the Responsive Boundary model is a form of Lagrangian polynomial interpolation. Lagrangian interpolation uses a number of discrete points to approximate a value of a function between these discrete points. In the Responsive Boundary model, the temperature is known at certain discrete locations, and it is desired to estimate the temperature at locations between these points. Constantinides and Mostoufi (1999) describe interpolation via Lagrangian polynomials thus:

$$P_n(x) = \sum_{k=0}^{n} p_k(x) f(x_k). \tag{3.19}$$

Here, P_n is the interpolated value using n basis points, p_k is the k^{th} Lagrangian polynomial, and f is the function value at the k^{th} discrete point, x_k. Note that $P_n(x)$ is evaluated at some point x between the discrete points x_k. The Lagrangian polynomials are defined:

$$p_k(x) = \prod_{i=0,\ i \neq k}^{n} \left(\frac{x - x_i}{x_k - x_i} \right). \tag{3.20}$$

Lagrangian interpolation can use any number of points, but care must taken not to use too many so as to avoid highly erratic behavior of high order polynomials between the node points. In the case of the Responsive Boundary model, the number of nodes used can be adjusted. The code examines the desired position in the pool for interpolation and selects the closest four points as a basis for interpolation. The error associated with Lagrangian polynomial interpolation is also described by Constantinides and Mostoufi (1999):

$$R_n(x) = \prod_{i=0}^{n} \big(x - x_i\big) \frac{f^{n+1}(\xi)}{(n+1)!},\ x_o \leq \xi \leq x_n. \tag{3.21}$$

It is important to note that the error in interpolation decreases with increasing n as long as the polynomials are not too high of order to cause a large degree of fluctuation between

the basis points.

Another quantity of interest is the pool's surface temperature. In this case, instead of interpolation, extrapolation is required. Initially the surface temperature was taken to be the liquid temperature of the top node. There are a couple of problems with this. First, the temperature of the top node is not technically the surface temperature, it is the temperature of the liquid half-way through the top node. Second, the temperature of the top node is taken from a location that changes with the the spatial step size, so the location of the surface temperature changes as the liquid pool shrinks, and it is impossible to compare surface temperatures between simulations that use different spatial step sizes, and that makes grid convergence analysis of the surface temperature and therefore the mass burn rate impossible. Instead, it was decided to estimate the surface temperature at the surface using extrapolation of nearby temperatures. At first, Lagrangian polynomials were used, but the error from this method led to obviously erroneous values for the surface temperature. For example, in some cases the projected polynomial predicts a surface temperature that is lower than the temperatures at the nodes immediately below the surface. Instead, the temperature profile near the surface is assumed to conform to the model represented by Equation 2.7. The exact form of the equation used for temperature in this case is:

$$T(z) = k_1 e^{-k_2 z}. \tag{3.22}$$

Equation 3.22 uses two parameters, k1, and k2 which are determined from two temperature data points. The data points come from the top two nodes nearest the surface. With the two parameters determined, the application of the temperature model yields an estimate for the surface temperature that is far more consistently with the physical system than the predictions given from Lagrangian extrapolation. With the two top temperature nodes, the Responsive Boundary model can extrapolate a value for the surface temperature that is consistent in location and comparable between simulations of different spatial mesh resolution. Figure 3.4 demonstrates the top of the liquid pool domain and shows the locations of the nodes involved in the extrapolation.

3.6 Overview of the Responsive Boundary Code

The Responsive Boundary model is written in C++, which is an object-oriented language. Such languages are based on classes with contain specific data members and methods. Since the Responsive Boundary model is designed to be incorporated with the ARCHES fire code, a similar programming style is used with the boundary model that is based on that of ARCHES. Each class in ARCHES is defined with two files. The first is a header file ([classname].h), and this file contains the definition of the class and the declaration of all its methods and members. The second file is a .cc-file, and it contains the specific coding for all the major methods in that class. The Responsive Boundary model uses this form and consists of three classes: The ResponsiveBoundary class, the RBMixerProperties class, and the RBComponentProperties class.

The RBComponentProperties class contains information relating to the physical properties of a pure fuel species. Indeed, this class is designed to model a pure chemical species. It's members are the identity of the fuel itself and several basic properties such as its critical temperature, pressure, volume, and its acentric factor. It also includes information as to whether it is a paraffin, a cyclo-paraffin, an aromatic, or an alcohol. Such data are necessary for the computing of the component's physical properties which is a task accomplished by the methods of the RBComponentProperties class. Examples include the calculation of the pure species heat capacity as a function of temperature, gas and liquid phase viscosity, liquid density, liquid and gas phase thermal conductivity, and the vapor pressure. Currently there are 18 fuels programmed into the constructor of this class, and more fuels can be easily programmed as well. Some example include n-heptane, n-octane, n-dodecane, tetralin, decalin, toluene, benzene, xylene, methanol, and iso-propanol.

The RBMixerProperties class contains information on a fuel mixture, and it relies on information provided by the methods in the RBComponentProperties class. If the system of interest is a six-component fuel then the RBMixerProperties class will use six instances of the RBComponentProperties class as members. If the fuel is only single component then it will only use one instance the RBComponentProperties class to calculate the needed physical properties. The RBMixerProperties class uses physical properties of a mixutre as its members the same way the RBComponentProperties class does for pure components.

It's methods calculate properties such as the thermal conductivity, heat capacity, and density of a liquid mixture. It also calculates the properties of the gas film through which the volatile species will transport. The mass burn rate is calculated in this class as function of the temperature, pressure, and composition of the system. Activity coefficients and VLE data such as the bubble point temperature of a liquid mixture are computed by this class.

The ResponsiveBoundary class performs the time stepping calculations for the model. It keeps track of the primary variables like the liquid height, the liquid temperature profile, and the liquid composition, and it applies the fourth order Runga-Kutta routine for time stepping calculations. It also performs the calculations that account for the liquid drop and the interpolation of the new temperature values. It contains methods that can perform verification via MMS to test the code, and some of the methods in this class are designed to port the Responsive Boundary model with the ARCHES fire code. The ResponsiveBoundary class uses one instance of the RBMixerProperties class as a member to obtain the physical property data needed to solve the discretized continuum equations.

3.7 Interaction with ARCHES

Given the discussion of how the Responsive Boundary works so far, the question arises that if the model assumes a uniform temperature horizontally, how can it predict the behavior of a real pool where the temperature and mass burn rate vary across the surface, and if only one radiation input value is given to the Responsive Boundary model how can it account for the fact that feedback radiation to the pool surface varies widely with both space and time? The answer to this issue is that more than one instance of the Responsive Boundary model is used in its ARCHES application. In ARCHES the computational domain is three-dimensional. The domain and the location of the pool are illustrated in Figure 3.5.

ARCHES is primarily concerned with the gas phase reactions occurring over the pool. The pool is located at the bottom of the domain illustrated in Figure 3.5. The three-dimensional domain is divided spatially into three-dimensional nodes that fill the volume. Some of these nodes border the bottom of the domain where the pool boundary is located. For each node that is adjacent to the pool there is a small section of the pool surface as

indicated by the arrow pointing to such an area in Figure 3.5. For each of these areas on the pool surface an instance of the Responsive Boundary model is used to calculate the fuel velocity from that segment of the pool into the 3-d domain. Each pool area will give different inputs to the Responsive Boundary model, and each will get different values of the fuel velocity as output. The input required for the Responsive Boundary model from ARCHES include gas temperature just above the pool, fuel composition above the pool, radiation input to the pool surface, gas density above the pool surface, etc. The data needed to keep the various instances of the Responsive Boundary model across the pool surface include the liquid height, the liquid fuel composition, and liquid temperature profile. Such data are saved in maps and are associated with each area segment of the pool by the node location coordinates of the node just above that segment of the pool. While the code runs, these data are remembered and used with the appropriate instance of the Responsive Boundary model from time step to time step. If the code reaches its time limit and must be restarted, then the data needed for the Responsive Boundary model must be written to a file and read back to the code upon restart. ARCHES time steps its own calculations using its own methods. Once it completes a time step, the time step size is read into each Responsive Boundary model and they perform their calculations using that time step in order to keep pace with ARCHES. Generally, the time steps used by ARCHES are much smaller than those needed for the Responsive Boundary model, so only one time step is required from the boundary model, and at each time step the fuel inlet boundary condition is updated. In this manner, the Responsive Boundary model provides a fuel inlet boundary that varies with position over the pool and varies with time.

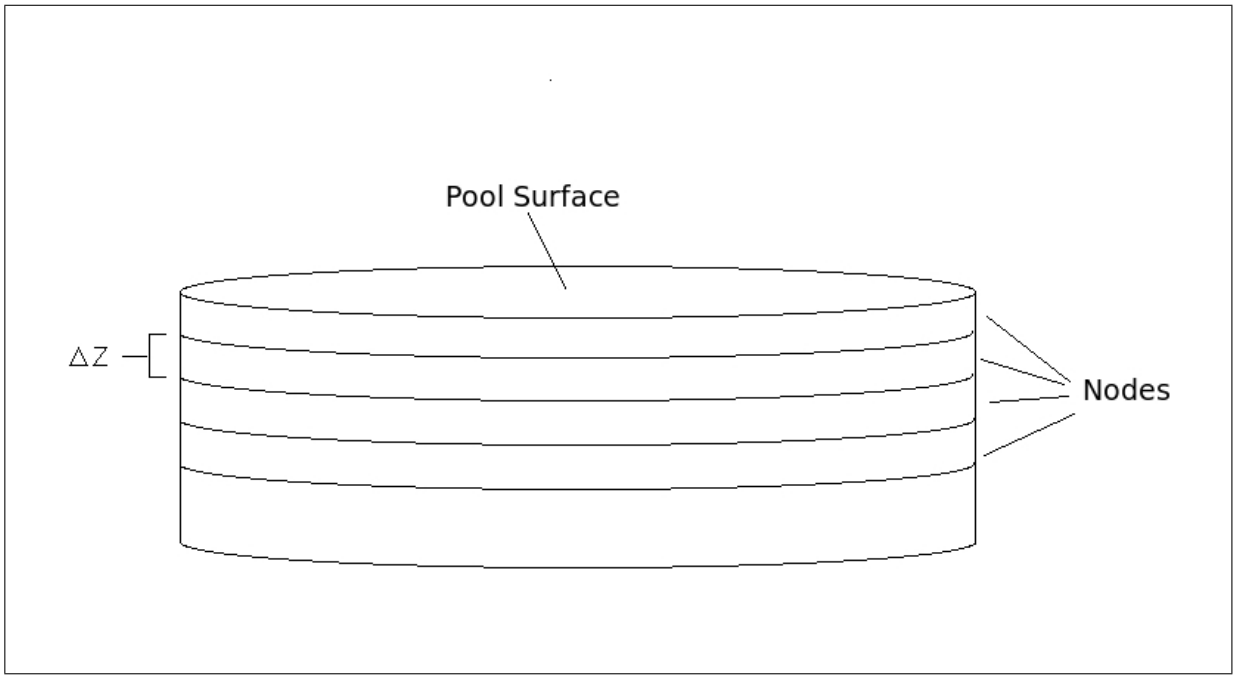

Figure 3.1: The computational domain of the liquid pool for the Responsive Boundary model.

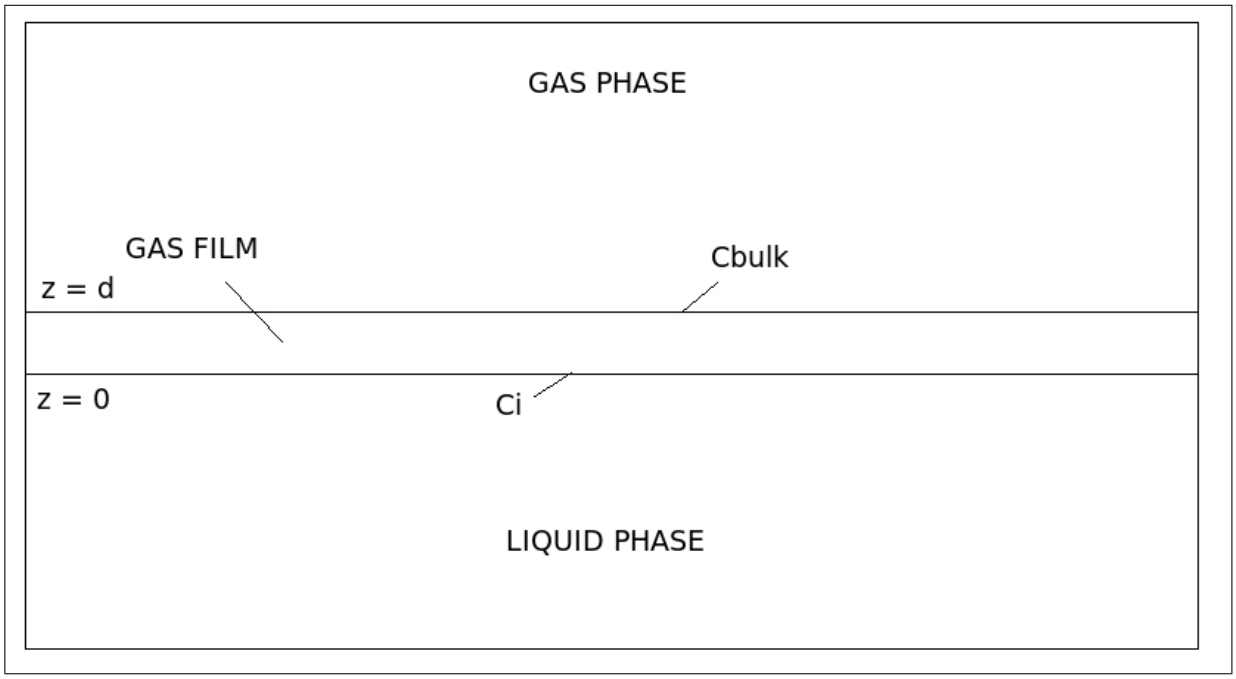

Figure 3.2: Fluid/fluid mass transfer according to film theory.

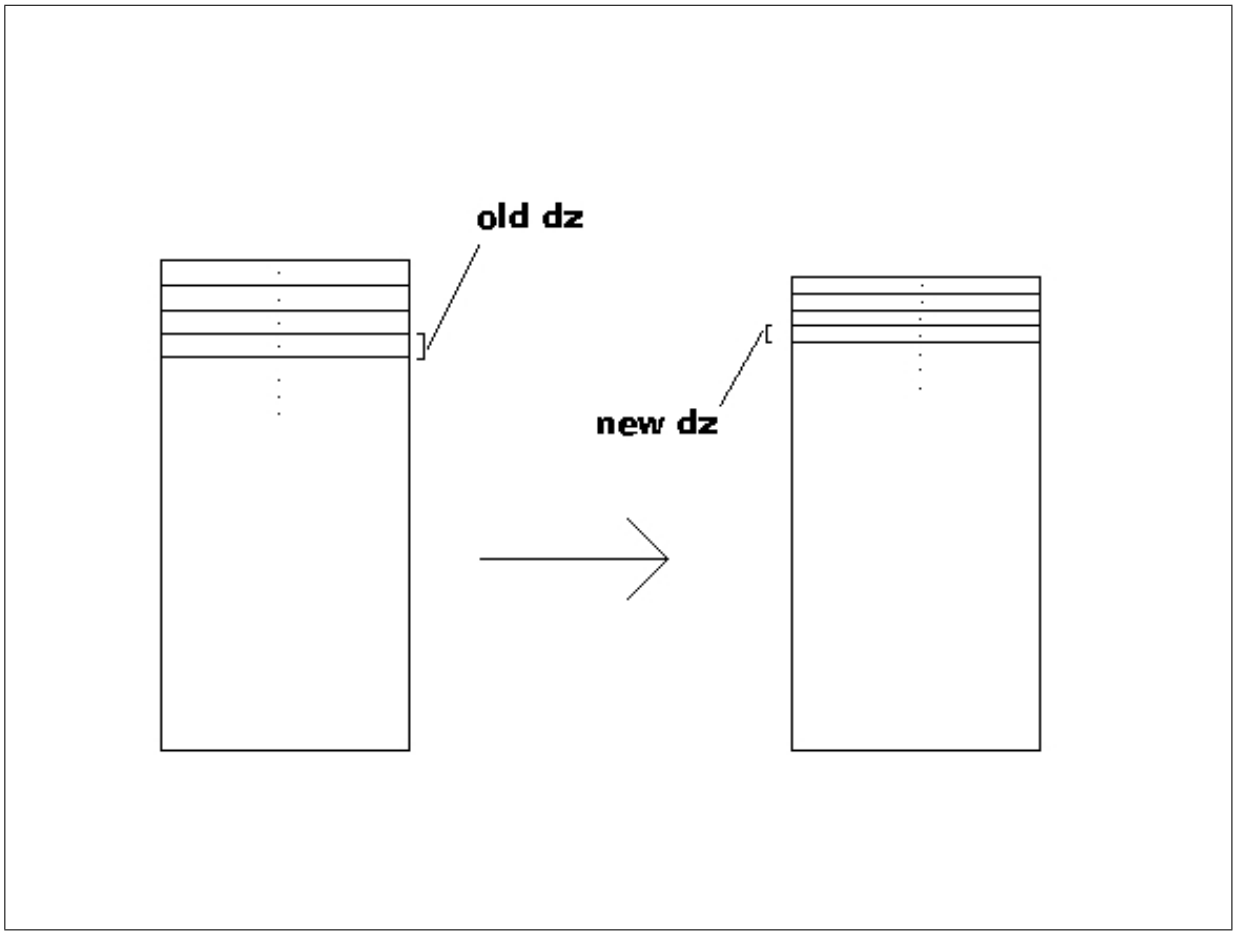

Figure 3.3: How the computational domain changes with liquid drop.

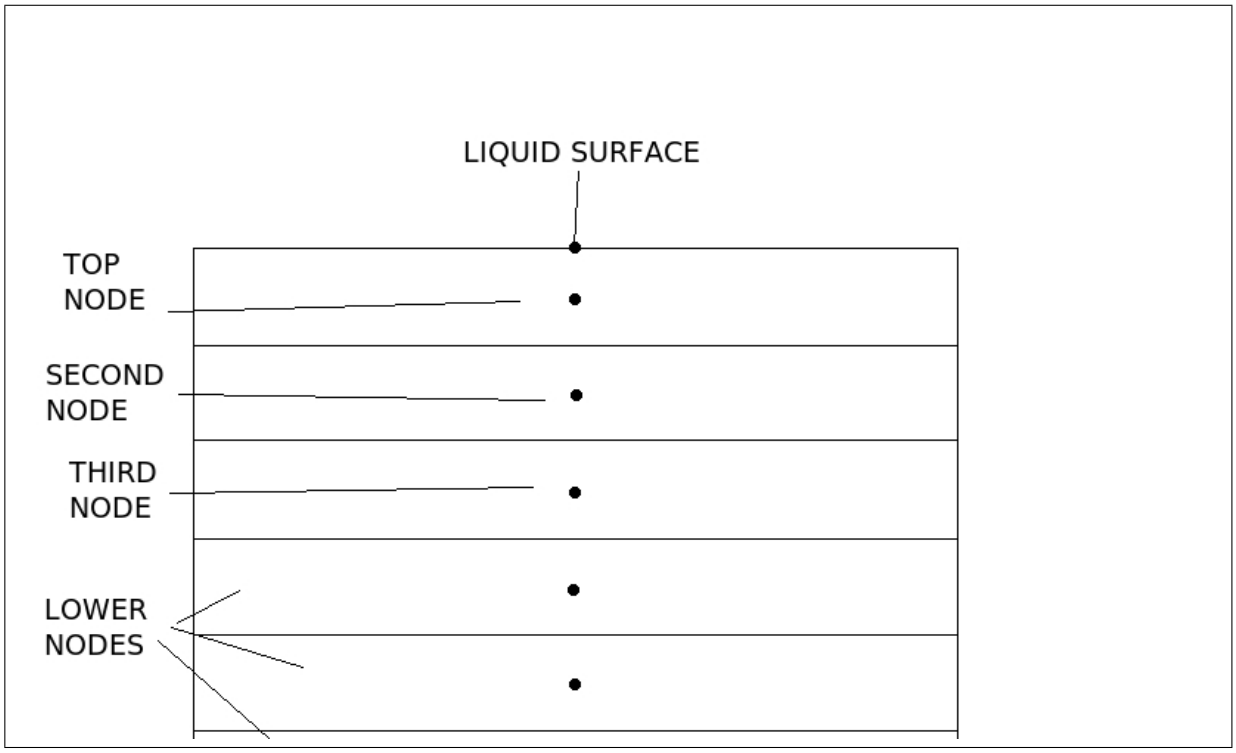

Figure 3.4: The top nodes of the pool domain and the location of the extrapolated surface temperature.

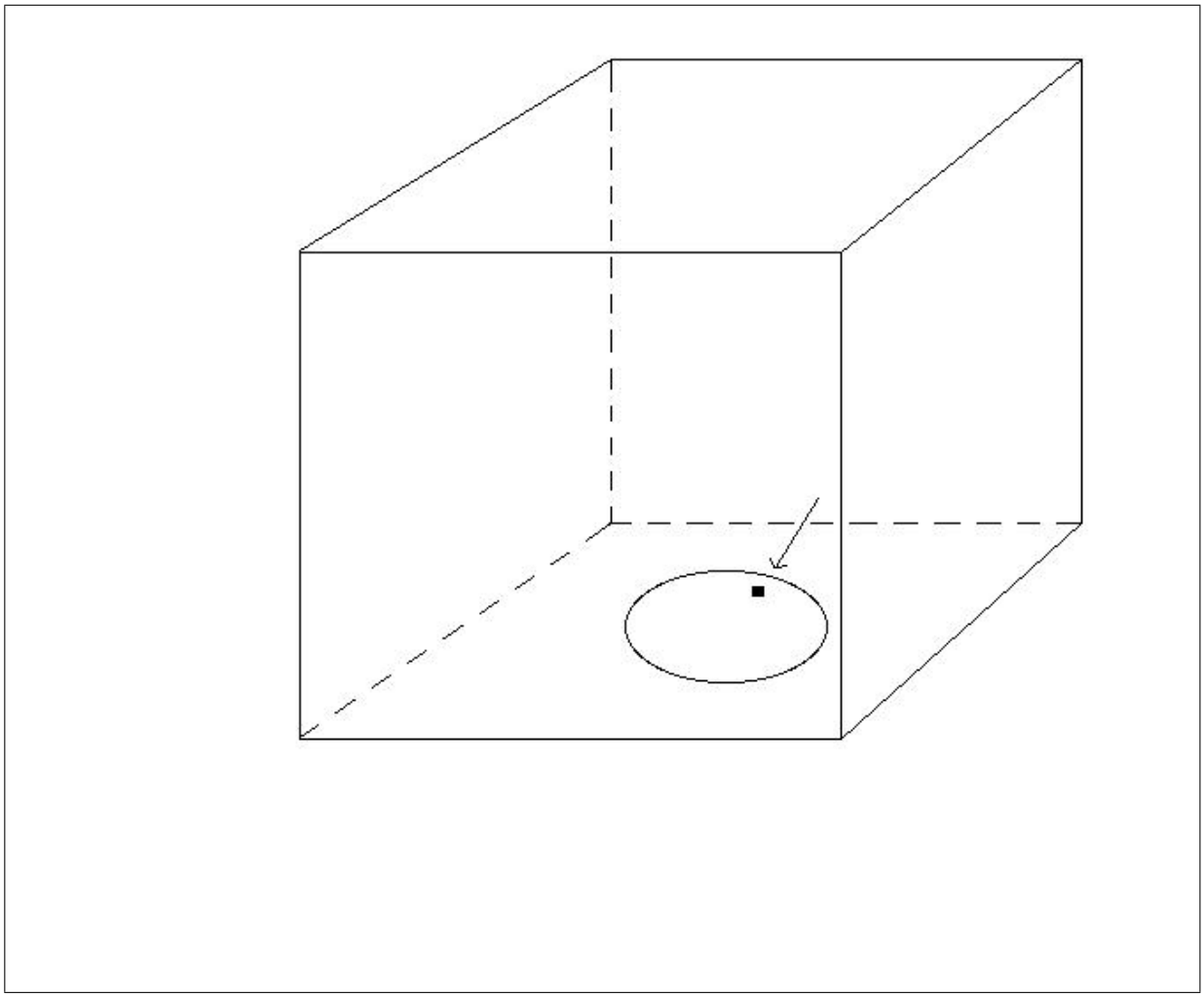

Figure 3.5: A depiction of the three-dimensional domain of a pool fire in ARCHES.

CHAPTER 4

VERIFYING AND VALIDATING THE RESPONSIVE BOUNDARY MODEL

4.1 Introduction

With the theoretical foundation for the Responsive Boundary model having been set down and the model itself having been described, the next step is to build confidence in its calculations through the methods of verification, validation, and uncertainty quantification as described in Chapter 2. As noted already, it is not possible to verify or validate a code, but rather the more meaningful objective is to verify and validate specific calculations of a code in order to establish confidence in the output of a model for given applications of interest.

In the case of the Responsive Model model, the applications of interest start with moderate to large sized, single-component liquid pool fires. The experimental data provided by Blanchat et al. (2006) and Klassen and Gore (1992) provide excellent comparisons for verifying and validating the Responsive Boundary model. The verification, validation, and uncertainty quantification for the Responsive Boundary model occurs in two phases. The model is first developed on its own and tested using given inputs suggested from experimental data to see if the model output is consistent with experimental results. Once sufficient verification and validation is performed on the model by itself, the model is then linked to the gas-phase fire code, and verification and validation tests are performed on ARCHES simulation output. The purpose for performing code analysis in this manner is to focus first on the Responsive Boundary code first and resolve any issues in that smaller environment before linking the model to a much larger and more complicated fire code. This chapter deals with the verification, validation, and uncertainty quantification activities with the Responsive Boundary model on its own. The next chapter will discuss these activities for the Responsive Boundary linked with ARCHES.

It is important to note that there are additional limitations to how much one can test the performance of the Responsive Boundary model on its own. As currently written, the model requires certain inputs to be able to give the appropriate prediction of the mass burn rate. Without ARCHES, there are limitations on the accuracy and availability of these inputs. These inputs include incident radiation flux to the pool surface, flame temperature over the pool surface, and fuel compositions over the fuel surface. Each of these properties varies widely over the spatial locations of the pool surface. For the purposes of testing the Responsive Boundary code only one value for each of these can be used as input to test the pool code by itself. The objective is to see if the model will return a reasonable, average output (i.e., mass burn rate) with given inputs that are averaged over time and over the pool surface. In the case of the vapor phase fuel composition over the pool surface, the value is set to zero in order to avoid certain complications. This is done with the understanding that the predicted mass burn rates will be higher than those expected from the model when used with ARCHES.

4.2 Verification: Manufactured Solutions

The method of manufactured solutions is an effective means to discover subtle bugs in the code that do not appear in compilation, cause a crash in the execution of the code, or result in obvious error in the results. It is effective in determining if the code is actually solving the continuum equations correctly, and it is useful for determining if the error in the solutions to the continuum equations are what they should be. In other words, the method can determine if the discretization error is of the correct order.

Two manufactured solution are chosen to test the ability of the Responsive Boundary code to solve the energy balance (Equation 3.2) and the height equation (Equation 3.4). These equations form the core of the Responsive Boundary model for simulations of single-component fuels and must be verified. The temperature profiles resulting from the solution of the energy balance equation change with depth in the liquid and with time. The height of the pool only changes with time. The manufactured solutions for these two quantities need to have the same dependencies with space and time. For verification of the energy equation, the domain is given a length of one meter, and all liquid evaporation and therefore

change in the domain size is suspended. The chosen solution is a simple sinusoidal function whose amplitude begins at a value of six and shrinks exponentially with time to a value of one. The values of the manufactured solution are chosen to be similar to the temperatures commonly produced by the energy equation in order to avoid any problems from the property calculations that depend on temperature. For example, the code is not likely to produce a real value for the viscosity of liquid JP-8 if the input temperature from the manufactured solution is -100 Kelvins. This would cause the code to crash and defeat the purpose of the manufactured solution in the verification process. To meet these criteria the manufactured solution for the energy equation is:

$$T(z,t) = T_{Nominal} + \left(1 + 5e^{-t}\right) \sin\left(2\pi z\right). \tag{4.1}$$

The solution oscillates about $T_{Nominal}$ which is assigned as value of 298.15 Kelvins in the code. The solution given in Equation 4.1 is easily differentiated in both time, t and liquid depth, z. For verification of the height equation the energy balance is turned off and the temperature values are given a constant value across the domain of the liquid and do not change with time. The manufactured solution for the height equation has similar character to the height of an evaporating pool. Specifically, it starts at an initial value, in this case a value of 1, and then it shrinks with time. The manufactured solution is expressed:

$$H(t) = 1 - \ln\left(1 + \alpha t\right),\ \alpha = \frac{e-1}{t_{end}}. \tag{4.2}$$

The function described in Equation 4.2 at the initial time starts with a height of 1. Then the height drops logarithmically to a value of 0 when time is at t_{end}.

The manufactured solutions described in Equations 4.1 and 4.2 are applied separately to the Responsive Boundary model to test the solution of the energy and height balance. If the code works they way it is intended it will produce the manufactured solutions with some degree of error. This error stems mostly from the discretization of the continuum equations. Several potential bugs that would otherwise be difficult to detect could cause additional error to the model's output of the manufactured solution to the point that the

output would have none of the character of the manufactured solution. If the output from the model matches the character of the manufactured solution, the next step is to examine the error in the output at different mesh resolutions. A plot of the log of the error against the log of the step size produces a line with a slope equal to the order of the discretization error with respect to the step size.

For the energy equation, a discretization error of second order is expected with respect to spatial step size, and a fourth order error is expected with respect to the time step size. For the height equation a fourth order discretization error is expected with respect to time step. A Responsive Boundary simulation is run with the manufactured solution source term for the energy equation in effect. The simulation is run on a one-meter domain with a spatial step size of 0.005 meters and a temporal step size of 0.02 seconds. The simulation runs for five seconds. The output from the model is shown in Figure 4.1. A semi-logarithmic plot of the relative error for the same simulation is shown in Figure 4.2. In both figures the output for three different times (one, three, and five seconds) from the simulation is plotted. Examination of Figure 4.1 shows that the solution does indeed look like a sine wave from 0 to 2 whose amplitude shrinks with time. The error plot in Figure 4.2 also shows that the relative error for the whole domain is quite small, on the order of 10^{-11} to about 10^{-13} in the center of the domain. The error rises to about 10^{-5} near the boundaries. This is due to the proximity to approximate boundary conditions. As stated in the previous chapter, the diffusive flux is assumed to be zero at the boundaries. The rise in error may also be attributed to the form of the discretization of the derivative term at the boundaries. Since the thermal diffusive flux into the domain at the boundary is assumed zero, the derivative term collapses from a second order form like that shown in Equation 3.9 to a first order form. It is also interesting to note that as time proceeds in the simulation the error increases somewhat. This is likely discretization error accumulating with each subsequent time step.

Another simulation is run with the Responsive Boundary model using the manufactured solution for the height equation. This simulation uses a time step of 0.02 seconds. Figure 4.3 shows the plot of the output from the model for the height of the liquid with time, and Figure 4.4 shows the relative error in the model's output. The manufactured solution

for this simulation, described in Equation 4.2 uses an end-time parameter (t_{end}) of 50 seconds. This means the liquid level starts at one meter and falls to zero in 50 seconds. The simulation is run for 30 seconds. Examination of Figures 4.3 and 4.4 show that the output from the model matches the character of the manufactured solution closely just as it did for the manufactured solution for the energy equation, and the error from the output is very small (10^{-14} or less.) Also, as with the first MMS simulation the discretization error accumulates with time.

4.2.1 Grid Convergence Tests with the Manufactured Solutions

The results shown in Figures 4.1 to 4.4 justify a certain level of confidence that the code is solving the continuum equations for energy and height correctly within an acceptable level of discretization error. There is cause for confidence that the code is free of certain types of errors. However, while it is known that the code solves these equations with relatively small error, it remains to be determined if the discretization error produced in the code is of the expected order. To determine the order of error it is necessary to run several manufactured solution simulations like those described above at different temporal and spatial step sizes and examine the trends in the error.

First, for the energy equation, several simulations are run with the manufactured solution at different spatial step sizes. The step sizes used are $\frac{1}{10}$, $\frac{1}{30}$, $\frac{1}{90}$, $\frac{1}{270}$, and $\frac{1}{810}$ of a meter. A time step of 0.01 seconds is used in each simulation. The simulations run for 6 seconds of simulation time and values of temperature at 6 seconds were analyzed. The temperature values from the manufactured solution are examined at different points in the domain. The log of the error is plotted against the log of the spatial step size for the manufactured temperature at a depth of 0.25 meters below the pool surface, and the plot is shown in Figure 4.5. Examination of the plot shows that a straight line is produced with a slope of 2. This suggests the order of discretization error with spatial step size for the energy equation is 2, as expected. Lines with slopes of one and two are also provided in Figure 4.5 for comparison with the data. Similar analysis at different depths of the pool showed similar results.

To verify the order of discretization error with time step, another series of simulations using the manufactured solution for the energy equation are performed. The simulations

use time steps of 0.5, 0.25, 0.125, 0.0625, and 0.03125 seconds. Each simulation employs a spatial step size of $\frac{1}{800}$ of meter, and runs for 6 seconds. Once again, the log of the error in the output for each simulation is plotted against the log of the time step. The plot is shown in Figure 4.6. Once again, the data are taken from a depth of 0.25 meters below the pool surface at a simulation time of 6 seconds. Examination of the data in Figure 4.6 shows a straight line with a slope of 4. This suggests that the discretization error in the energy equation with respect to time step is fourth order, as expected.

Figure 4.7 shows another plot of the log of the error in simulation output against the log of the time step. These data arise from a series of simulations where the manufactured solution for the height equation are applied. The simulations use time steps of 1.0, 0.5, 0.25, 0.125, and 0.0625 seconds, and they each run for 30 seconds. The height at 30 seconds is used as the model output in the grid convergence plot in Figure 4.7. Figure 4.7 shows a straight lines with a slope of 4. Once again this demonstrates that the values of height for the manufactured solution from the model have a discretization error with respect to time step that is fourth order, and this is consistent with the use of a properly programmed Runga-Kutta fourth order method. The results illustrated in Figures 4.5, 4.6, and 4.7 demonstrate that for the given manufactured solutions the code not only gives results that are apparently correct from inspection, but that the results produce the errors they should given the discretization methods used.

4.3 Verification: Grid Convergence of Model Output and Grid Convergence Index

The exercises described in the previous section suggest that the time stepping algorithm and spatial approximation used in the Responsive Boundary model function as they should without significant coding bugs. However, the manufactured solutions described do not test every aspect of the code. The manufactured solutions chosen for the energy and mass balance are applied and tested separately. The full implementation of the Responsive Boundary model applies both equations in a coupled manner, and the code applies interpolation and domain shrinkage to mimic the behavior of an evaporating pool. The manufactured solutions provide a starting foundation for verification of the Responsive Boundary model's output, but more must be done to verify the computations of the model.

The next set of exercises test the code directly with all coupled equations being solved at the same time and with domain shrinkage and interpolation in tact. Grid convergence is applied again to the results of the model, only this time there is no manufactured solution. The tested data are the actual intended output of the model which is the mass burn rate of fuel. The advantage to this method is that the whole model is tested in its intended use. Any errors arising from the coupling of the continuum equation, the shrinking domain, or from interpolation error are likely to be detected from these grid convergence studies. The disadvantage to these exercises is that there is no direct means of measuring error in the model's solution compared to the actual model continuum solution. The error must be estimated from the Grid Convergence Index technique described in Chapter 2.

The objective for these exercises is similar to that for the manufactured solutions in the previous section. The order of discretization error must be verified for the energy and mass balance equation for the Responsive Boundary model running in its intended application, and the discretization error must be estimated using the GCI once the order is verified. Unlike the grid convergence tests applied with the manufactured solution simulations, more than two grids must be used to calculate the order of error with respect to the spatial and temporal step sizes. In this set of exercises several simulations are run at different mesh sizes, and the appropriate output from each simulation are compared to calculate the order of convergence using Equation 2.34. Once the order of convergence is verified, the discretization error is estimated using the GCI (Equation 2.33). Since the order of error is confirmed a value of 1.25 is applied for the parameter F_S in Equation 2.33. Once again the output of the simulation must be tested to determine the order of error with respect to the spatial and temporal time steps. The output for the model is the mass burn rate of the fuel. If the code operates as it is designed the output should have discretization error that is second order with respect to spatial step size and fourth order with respect to time step size, just as with the previous grid convergence tests.

For all verification tests described in this section the simulation is based on the conditions of a low wind, JP-8 pool fire experiment conducted by Blanchat et al. (2006). The input parameters include a pool diameter of 7.93 meters, an average wind speed of 0.85 m/sec, and an average radiative heat flux of 85,100 W/m^2 to the pool surface. Also, as

already stated, the fuel vapor composition over the pool is not known from experiment and is set to zero for the purposes of verification and validation. The presence of fuel in the vapor inhibits the burn rate, so mass burn rates calculated in these exercises will be marginally higher than they would be in the ARCHES framework where the vapor composition is available.

The first two tests examine the convergence of liquid temperature and mass burn rate with spatial step. Tables 4.1 and 4.2 show the results of these tests for the liquid temperature and mass burn rate, respectively. For these tests the average flame temperature over the pool is set arbitrarily to 1000 K. The flame temperature is another quantity that ultimately must be provided from the fire code in ARCHES. Examination of the data in Tables 4.1 and 4.2 shows the grid sizes used in each test, the values computed, the calculated orders of convergence from Equation 2.34, and GCI for the finest grids from Equation 2.33. To calculate the order of accuracy from Equation 2.34 requires at least three grid sizes. In these tests as many grid sizes are used as is possible. For the liquid temperature verification, five grid sizes are used. The simulation uses a starting domain size of 40 centimeters of pool depth, and a time step of 0.01 seconds is used. The tested temperature value is taken from a height of 34 centimeters (6 centimeters below the surface.) Since five grids are used, three estimates of the order of error can be calculated. In Table 4.1 all convergence results are given for this series of simulations. The coarsest grid is assigned the highest case number while the finest grid is assigned a case number of 1. The symbol for the computed orders of error is assigned three numbers corresponding to the three cases whose values are applied to Equation 2.34 (i.e., $pijk$). As the grids used to compute the order of error become finer the value for the order of error should approach the asymptotic value. The error band according to the GCI for each of the finer grids is also given in Table 4.1 and in subsequent tables. Another means of determining if the computed order is approaching the asymptotic limit is suggested by Roache (1998). He suggests that the asymptotic limit is reached if the ratio of the finer GCI to the coarser GCI approaches a value of r^p, where r is the refinement ratio and p is the observed order of accuracy. For the case of the liquid temperature grid convergence test shown in Table 4.1, the refinement ratio used is shown there, and the observed order of error is 2. This means that a GCI ratio of nine (labeled as

target ratio in the table) is expected for the asymptotic region, and this is indeed the case between cases two and one in Table 4.1. The different values of GCI (labeled as ″Error Band″) expressed in Table 4.1 are given as fractions as opposed to percents. Either way the estimated error bands suggested by the GCI here are quite small.

Table 4.2 is a similar table for a series of simulations that test the convergence of the mass burn rate with spatial step. In this series of tests an initial domain height of 56 centimeters is used. The simulations run for 30 seconds, and they each apply a 0.01 second time step. In this series of tests six simulations using six different grid sizes with a refinement ratio of two are applied. This gives four calculations for the order of discretization error and four calculations for the error band for the four finest grids. Examination of these data show that as the grid becomes finer the ratio of error band range approaches the expected value for the asymptotic range in the calculated order (in this case a value of 4.) Also note that the error band for this quantity is larger than that for the liquid temperature, so the error band range is expressed as a percentage. The data for these two verification exercises as shown in Tables 4.1 and 4.2 suggest that the order of discretization error for both the liquid temperature and the mass burn rate is two with respect to the spatial step size. This is the expected value for these quantities since the discretization of the energy equation is second order with respect to spatial step size, and since the mass burn rate is a function of the liquid surface temperature, it too is expected to be second order. Due to the functional dependence the mass burn rate on liquid temperature, the mass burn rate is analyzed in the remaining grid convergence exercises instead of the liquid temperature.

The next two grid convergence exercises seek to test the order of discretization error with respect to time step size for the liquid height and the mass burn rate. Because the temporal discretization scheme for both the energy and height equations is fourth order, it is hoped that the grid convergence tests for both liquid height and mass burn rate will produce a fourth order error result. This series of tests uses an initial domain size of 0.50 meters, and a spatial step size of 0.001 meters. Each case runs to a simulation time of 30 seconds. Six cases are executed to give four calculations of the order of error and the GCI at the four finest grids. The coarsest time step is 0.5 seconds, and the finest is 0.015625

seconds. The data for mass burn rate and liquid height are shown in Tables 4.3 and 4.4, respectively. Examination of the data in Tables 4.3 and 4.4 shows that for both the liquid height and the mass burn rate the order of discretization error converges to 1. This is not expected because the Runga-Kutta routine used in these simulations is fourth order with respect to time step size. The ratio of GCI does converge to the target value for a first order error so there is confidence that the order calculations in Tables 4.3 and 4.4 are in the asymptotic convergence range, and the computed error is indeed first order with time step. This result is different from that given from the grid convergence tests with the manufactured solutions. Since the tests on the manufactured solutions indicated fourth order error with time step it is reasonable to conclude that there is nothing wrong with the coding of the Runga-Kutta time step routine. Rather, there must be a process involved in the full Responsive Boundary simulations that was not present in the manufactured solution simulations. The most likely suspect is the interpolation of liquid temperatures between the Runga-Kutta steps. The interpolation introduces another source of error to the liquid temperature values. Since mass burn rate is a function of temperature, the error would propagate to that quantity, and since the mass balance equation depends on the mass burn rate, the error would also propagate to the liquid height values. This extra source of error could cause these quantities to exhibit error of lower order than the expected value of 4. To test this idea, another series of grid convergence simulations is run. However, this time the interpolation step is removed by setting the right-hand side of Equation 3.4 to zero. This means that the liquid level will remain constant at its initial height for the whole simulation. The mass burn rate is still computed but it cannot impact the liquid height. The heat input simply warms the pool without vaporizing any liquid, and the simulation is set to run for 15 seconds. This is long enough to heat the surface of the pool substantially without the allowing the liquid to reach its boiling point. The depth of the liquid is 50 centimeters and the spatial step is 0.001 meters. Table 4.5 shows the time steps used in this grid convergence study as well as all the pertinent results. Results for the mass burn rate are shown but not for the liquid height as this quantity does not change for this set of simulations. Examination of Table 4.5 shows that without the interpolation step the calculated order of error for the mass burn rate with time step size is now the expected

value of 4.

The results of this series of simulations confirm that the additional error from the interpolation step reduces the order of error in the model's output from fourth order to first order with respect to time step. Numerous attempts were made to diminish the interpolation error and recover fourth order error by using different interpolation techniques. Initially, linear interpolation was used. When the lower order of discretization error was discovered, two varieties of cubic spline interpolation were tried, and when that failed, a higher order Lagrangian interpolation technique was attempted with no success. The order of error with time step was still first order. Although disappointing, this is not fatal to the performance of the Responsive Boundary model. A first order discretization error is still consistent. Consistent in this context means that for arbitrarily small time step sizes the discretization error due to time stepping approaches zero, and the mass burn rate predicted by the model approaches the value given by the true solution to the continuum equations.

The data from these verification exercises provide a sufficient foundation of confidence that for simple, single-component pool fire simulation the Responsive Boundary model's computer implementation provides a reasonable approximation to the solution of the continuum model. The order of discretization error for both spatial and temporal time steps is consistent, and an appropriate error band can be produced for the total discretization error. According to Roache (1998) the total error band can be computed by summing the GCI for the time step with that for the spatial step. For example, for the mass burn rate if a spatial step size of 1.17×10^{-4} meters (0.117 millimeters) is used, and a time step of 0.015625 seconds is used, according to Table 4.2 the error from spatial discretization would be $\pm 0.158\%$, and according to Table 4.3 the error from temporal discretization would be $\pm 0.312\%$. This would give an overall error band of $\pm 0.470\%$ in the mass burn rate given by the Responsive Boundary model due to the discretization of the continuum equations. In practice with ARCHES, the temporal discretization error band would be much smaller as the time steps utilized by the Responsive Boundary model are taken from the ARCHES time steps, and in ARCHES the standard maximum time step is set to 0.01 seconds, and the actual step size is generally much smaller.

4.4 Validation: Sensitivity Analysis

The last sections in this chapter describe the methods used to validate the calculations that the Responsive Boundary produces, and they describe to some degree the error caused by the assumptions in the conceptual model for the liquid pool fire. As already mentioned the primary output from the Responsive Boundary model is the mass burn rate from the pool surface. To generate that output a number of inputs are required for the model. For some of these inputs knowledge of their value comes from experiment with a degree of uncertainty. Other inputs are not measured directly in experiment and are input to the Responsive Boundary model from the fire code's calculations. At this preliminary stage of validation the latter type of inputs must be guessed based on experience with simulation in order to conduct a simple sensitivity analysis.

The purpose of the sensitivity analysis is to measure the range of responses from the model's output given a range of values from the various inputs. The exercise can also give indications of how robust the model is to various ranges of input. For example, as discussed in Chapter 2, pool fires composed of a single-component fuel have been shown to heat to a temperature slightly less than the fuel boiling point. In the analysis of the model it is of interest to see under what conditions the model predicts boiling for the single-component fuel contrary to observation. This way the programmer comes to know which combination of inputs lead to physically unrealistic outputs from the model.

In ideal conditions the range of values of the output could be thoroughly tested with all the ranges of values from the various inputs using a well designed, statistical experiment. However, the focus of this chapter is to conduct verification, validation, and uncertainty quantification on the Responsive Boundary alone in order to prepare it as far as possible for use with the ARCHES fire code. As mentioned before, certain inputs to the responsive boundary ultimately come from the fire code and can only be guessed. As before the approach is to try to measure the average response of the model given a set of average inputs. Examples of these types of input include the flame temperature above the pool, the incident radiative heat flux to the pool surface, and the vapor composition above the pool surface. The vapor composition is once again set to zero for the purposes of the validation exercises with the understanding that the predicted output will be higher than

it should be.

The sensitivity analysis here is not a complex statistically designed exercise, but instead, it is a simpler linear response exercise that seek to measure the effect of each input alone on the output. The objective is to identify which inputs the output variable is most sensitive to. The exercise does not test the full range of possible output values. Such a rigorous exercise is conducted on the fully coupled ARCHES/Responsive Boundary model and is described in a later chapter. For now, this exercise identifies the inputs that most need to be tested in the more thorough analysis.

The output to be examined for the Responsive Boundary model is the steady mass burn rate of the fuel. The sample case used in this analysis is the same used in the grid convergence analysis and is provided by Blanchat et al. (2006). The pool fire is 7.93 meters in diameter and is composed of JP-8 fuel. Six inputs are chosen for the analysis: the average radiative input to the pool, the average flame temperature over the pool, the wind speed, the system pressure, the system temperature (the temperature of the surroundings and the initial pool temperature), and the reflectivity of the pool surface to incident thermal radiation. First, the nominal case is chosen about which all the inputs will be varied. Based on the data given by Blanchat et al. an average incident radiation value of $85,100\ W/m^2$ is used. There is little data on the flame temperature over the surface of the pool, and an average value is needed for the analysis, so a value of 1000 K is arbitrarily chosen. The wind speed is measured at an average value of 0.8 m/sec, and this value is adopted as nominal in the sensitivity analysis. The nominal value for the pool surface reflectivity is set to 0.07, and the nominal values for the system temperature and pressure are selected at 298.15 Kelvins, and one bar, respectively. When the nominal case is run to steady state, the nominal output is calculated to be 0.0818 $kg/m^2 - sec$. The measured value for this pool fire experiment given by Blanchat et a is 0.0580 $kg/m^2 - sec$. Since the fuel vapor composition is assumed zero in the model an overestimate is expected. There is also uncertainty in the flame temperature, the radiation input, and the wind speed. The predicted value is on the same order of magnitude as the measured value, and this is encouraging.

With the nominal values for the inputs and the output set, a range of values is selected

for each input. For each test all the inputs are set to nominal value except one which is varied over its selected range as simulations are conducted to obtain the corresponding range of output values. The procedure is performed for all six inputs mentioned. Table 4.6 shows the nominal conditions as well as the range of tested input values with the corresponding range of output responses. The input ranges are chosen to cover a physically realistic range. For example, the system temperature is chosen to cover the coldest winter conditions to the hottest summer day. Some of the inputs use a larger range than is likely to be seen in the physical world (the system pressure for example), but the present analysis need not be too rigorous as the primary objective is simply to identify the most sensitive inputs. Examination of the range of output values in Table 4.6 shows that the most sensitive inputs are the radiative input, the flame temperature, and the wind speed. Even with the generous input ranges given for the system temperature and pressure, the resulting range in output values is much smaller than those for the more sensitive inputs. The actual range of uncertainty for system temperature and pressure would be much smaller. The range given for surface reflectivity is more typically for what would need to be tested in a more rigorous sensitivity analysis. While radiative input and flame temperature are among the most sensitive inputs for the model they are quantities to be predicted by the fire code, and as such they would not be used directly in a more rigorous sensitivity analysis. This is also true of the fuel vapor composition. The most likely variable to be tested in a full sensitivity analysis for determination of uncertainty in the model output would be the system temperature and pressure, the pool reflectivity, and the wind speed. The choice of which inputs to test would depend on the experimental conditions and the experimental uncertainty in these quantities. For example, tests conducted indoors would likely have little variability in the wind speed, and this quantity would not as likely be included in the sensitivity analysis.

Along with the sensitivity analysis parameters illustrated in Table 4.6. Plots of the the output response to each input are included in Figures 4.8 to 4.13.

Examination of Figures 4.8 to 4.13 shows that for most input parameters there is a linear or near-linear relationship between the model output and the different inputs. One exception to this pattern is seen in Figure 4.9 at the far right of the plot where the mass

burn rate stalls then jumps at higher flame temperatures. This pattern is due to boiling of the liquid fuel as the burn rate goes from mass transfer controlled to heat transfer controlled. As previously discussed this phenomena is not observed for single-component fuel fires. This is the reason why higher flame temperatures were not tested in this analysis as boiling had already occurred in the tested temperature range. This is not likely to be an issue in the full ARCHES simulations because the regions of high flame temperature tend to be regions of lower radiative heat flux (around the edges of the pool). With the lower radiative input the flame temperature can go much higher without causing boiling to occur. Also regions of the pool surface that have lower flame temperature (the center of the pool for example) tend to have higher radiative heat fluxes.

Another exception to the linear behavior seen in most of the responses is found in Figure 4.10. The response to change in wind speed starts slowly and then jumps around 0.5 m/sec. This is due to the effects of forced convection beginning to dominate over that of buoyant convection as the wind speed increases. As forced convection tends to give higher mass transfer rates than that of buoyant convection the mass burn rate increases at a greater rate.

4.5 Comparison with Experiment

The last validation activity conducted with the stand-alone Responsive Boundary model is a series of comparison simulations. The purpose is to see if the model will generate reasonable output given a series of input conditions from experiments. Once again, this test is not a rigorous validation exercise as certain data are not available as input to the boundary model until it is linked with the fire code. Also, certain inputs to the model that normally would vary widely over the pool surface in the physical system are entered as average values. These properties include the radiation heat flux to the pool surface and the flame temperature over the pool surface. Again, the fuel vapor composition over the pool surface is assumed zero for these simulations, so the resulting mass burn rates will be over-predicted. For the radiative heat flux, an average value is chosen based on the experimental data, and for the flame temperature a value of 800 K is arbitrarily chosen for each case involving soot-forming flames, and a value of 1000 K is chosen for a case using

flames with no soot-forming behavior (a methanol flame).

Six experiments are used in this set of simulations from two research studies. The first study is conducted by Blanchat et al. (2006), the other is done by Klassen and Gore (1992). These two studies are chosen because of the detail they include in their reports on experimental conditions and more importantly, because they include measurements of the radiative heat flux to the pool surface along with the values of the global mass burn rate. Blanchat et al. conduct four large JP-8 pool fire experiments in outdoor conditions. Two of the four experiments are done under higher wind conditions while the other two experience lower wind velocities. The other two experiments are conducted by Klassen and Gore. These experiments are done indoors and include moderately sized pool fires using heptane and methanol. The pertinent data for these experiments are shown in Table 4.7. The heptane and methanol flames are simple, single-component fuels and easy to model, but the JP-8 fuel is a complex mixture. For these simulations a single-component surrogate is used to simulate this type of fuel. The surogate uses the properties of n-dodecane with a heat of vaporization that is consistent with measurements with JP-8 taken by Eddings et al. (2005). Also note from Table 4.7 that Klassen and Gore report two values for the steady mass burn rate. These two values arise from different heptane and methanol pool experiments. They attribute the presence of a heat flux gauge on the pool surface as the reason for the lower burn rate in one of the experiments.

The data summarized in Table 4.7 are applied as input to the Responsive Boundary model. The model is run for a simulation time of 60 seconds in each case. The mass burn rate is observed to come to a steady value and that value is compared to the experimental value. The results for all cases are shown in Table 4.8. As expected in each case the simulated steady mass burn rate is higher than the experimental value because the model does not take into account the presence of fuel vapor above the pool. It is interesting to note that some values were over-predicted higher than others. Other sources of error include the average flame temperature over the pool, the pool surface reflectivity, the error in wind measurements, as well as the error in radiation heat flux measurements. For example, the radiation heat flux measurement reported by Blanchat et al. (2006) for case one carries an uncertainty of $\pm\, 23,200\, W/m^2$ (about $\pm\, 27.2\%$.) Plots of the mass burn rate for all six

simulations are also included in Figures 4.14 to 4.19.

Examination of the data in Figures 4.14 to 4.19 shows similar pattern of behavior in the mass burn rate as the pool heats up. Note that the simulation predicts a warm up time between 25 and 30 seconds for these cases with the exception of the methanol case where the predicted warm-up time is longer. Unfortunately, the data provided here can give only a partial confirmation of how well the Responsive Boundary model performs. Until data from the model coupled with the fire code can be examined it is hard to determine if the output from the model will be consistent with experiment. The values shown in Table 4.8 are on the correct order of magnitude, and the trends are correct. For example, the higher wind cases from Blanchat et al. give the highest burn rates. The burn rate for the low wind cases from Blanchat et al. are over-predicted by nearly the same percentage. The larger pool fires give the highest burn rates, and the methanol fire gives the lowest burn rate, as expected. The over-prediction for the methanol fire is not as high as the other fires. This is likely due to the average flame temperature being only 1000 K. As methanol is a nonluminous flame, its flame temperature is likely higher than that for JP-8 or heptane, and because of the lower burn rates, the flame is likely to have contact with a larger area of the pool surface.

While it will be necessary to see how the model works fully coupled with ARCHES to know how well it performs, there are some preliminary data for the 30-centimeter diameter, heptane pool fire from ARCHES that give some indication as to how the flame temperature and fuel composition will behave. Figure 4.20 shows the flame temperature over the pool after nine seconds of simulations time, and Figure 4.21 shows the heptane fuel mass fraction over the pool surface at the same time. For both Figures 4.20 and 4.21 the reported data are taken from a distance of 2.25 millimeters over the pool surface (the location of the nearest computational node near the pool surface.) In this case nine seconds places the fire at steady burn conditions. Figure 4.20 shows that at steady burning conditions the evaporating fuel pushes the flames away from the pool except at the edges. The air temperature over most of the pool is much less than the 800 K assumed in the data comparison simulations described earlier, and the temperature at the edges of the pool where the flames still touch the surface is near 1400 K. The average temperature is likely

much less than the assumed 800 K. This is certainly one reason that the predicted burn rates in Table 4.8 are so high compared to experiment. The other likely reason for the over-predicted values is the assumption of no fuel above the pool surface. Figure 4.21 shows that this is clearly not the case. At steady burning conditions there is over 90% fuel over the majority of the pool surface. Since at steady burning the pool surface temperature is near boiling point the fuel will still be able to vaporize, but it will be at a much lower rate. Whether or not the rate predicted by the coupled Responsive Boundary model is consistent with experiment is the subject of a later chapter.

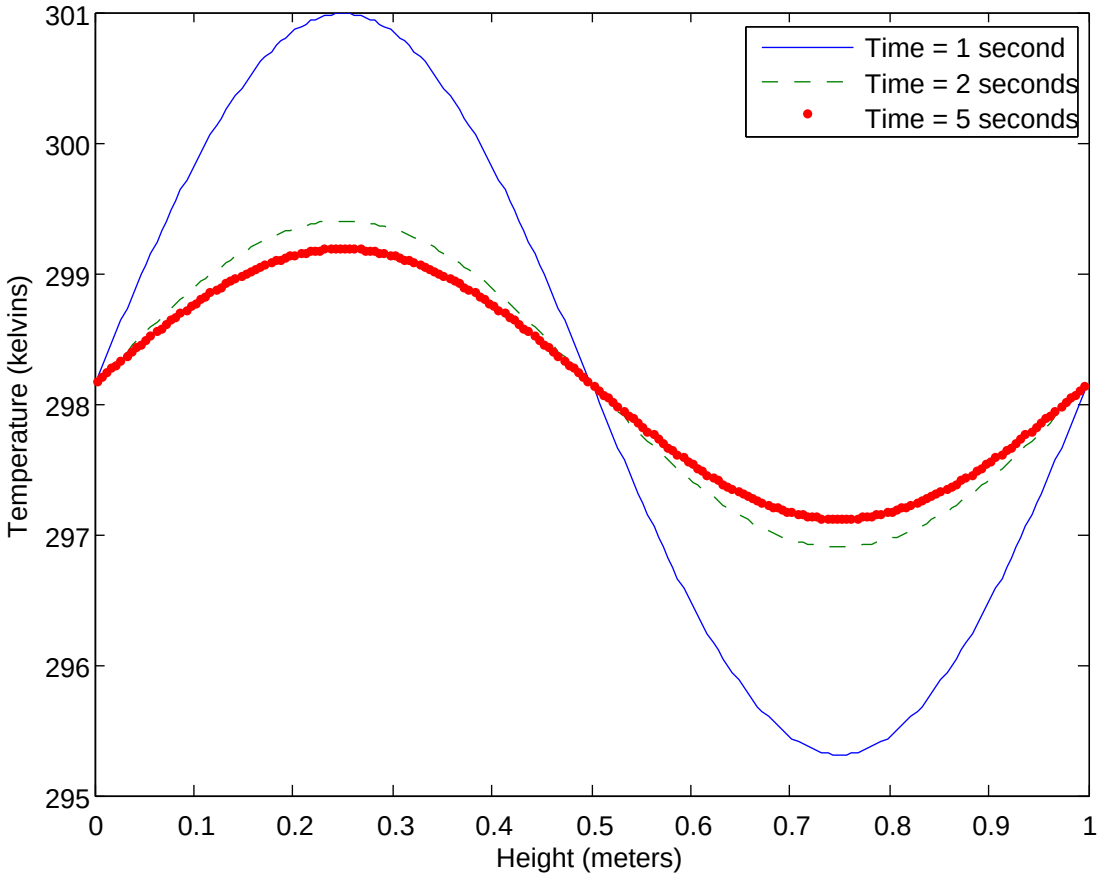

Figure 4.1: The output from the Responsive Boundary model for a manufactured solution to the energy balance equation.

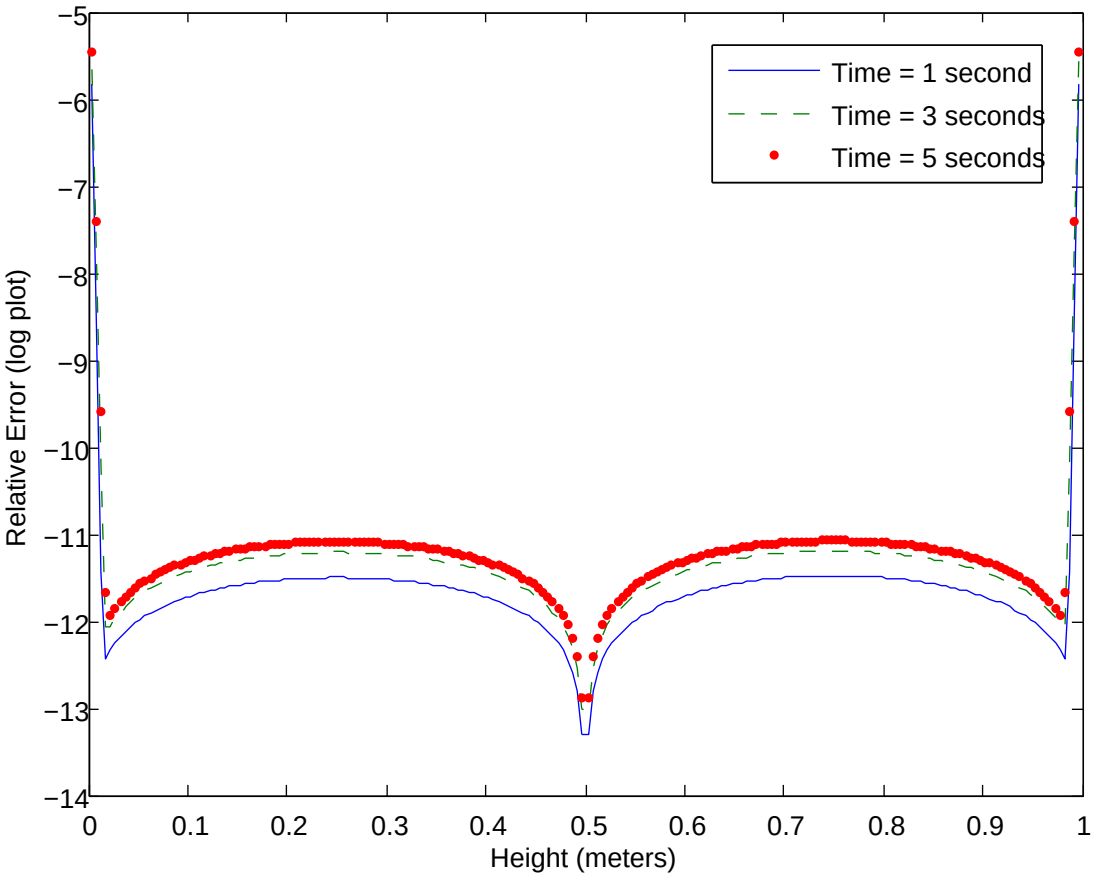

Figure 4.2: The relative error in the output from the Responsive Boundary model for a manufactured solution of the energy equation.

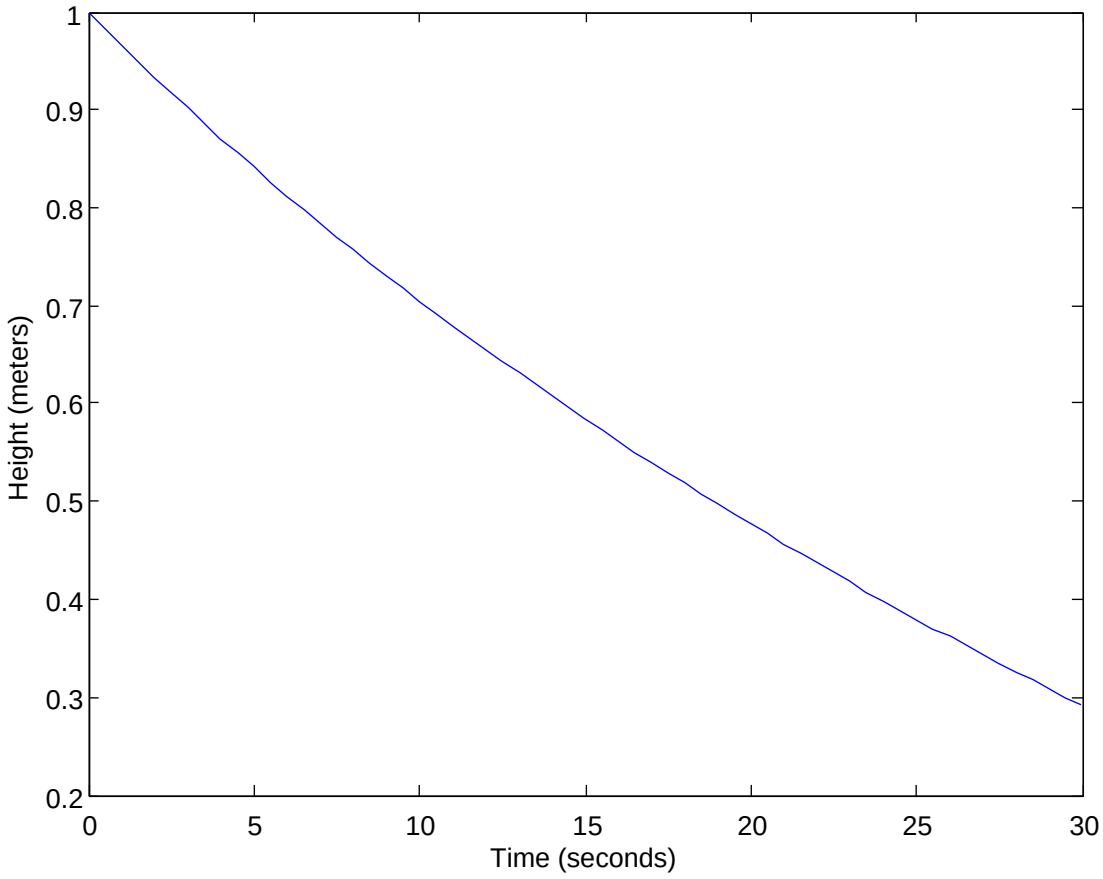

Figure 4.3: The model output for the manufactured solution for the height equation.

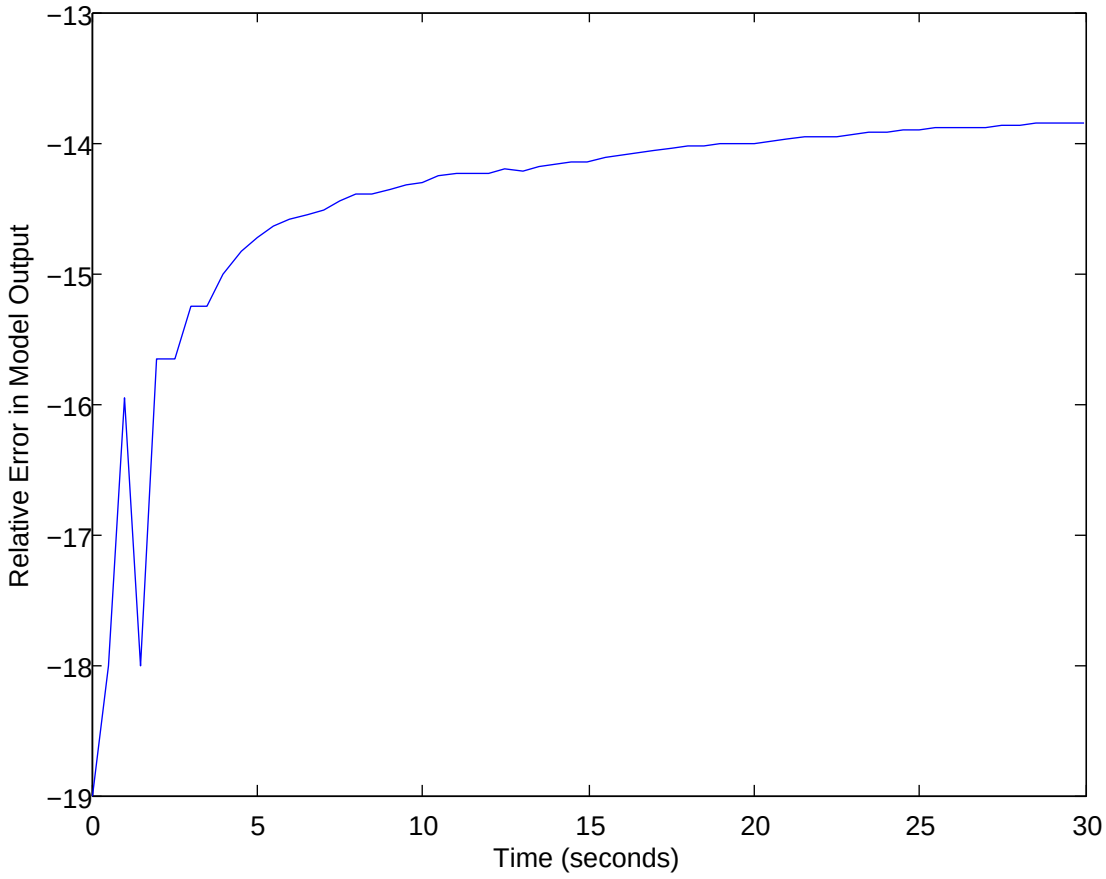

Figure 4.4: A log plot of the error in the model output for the manufactured solution for the height equation.

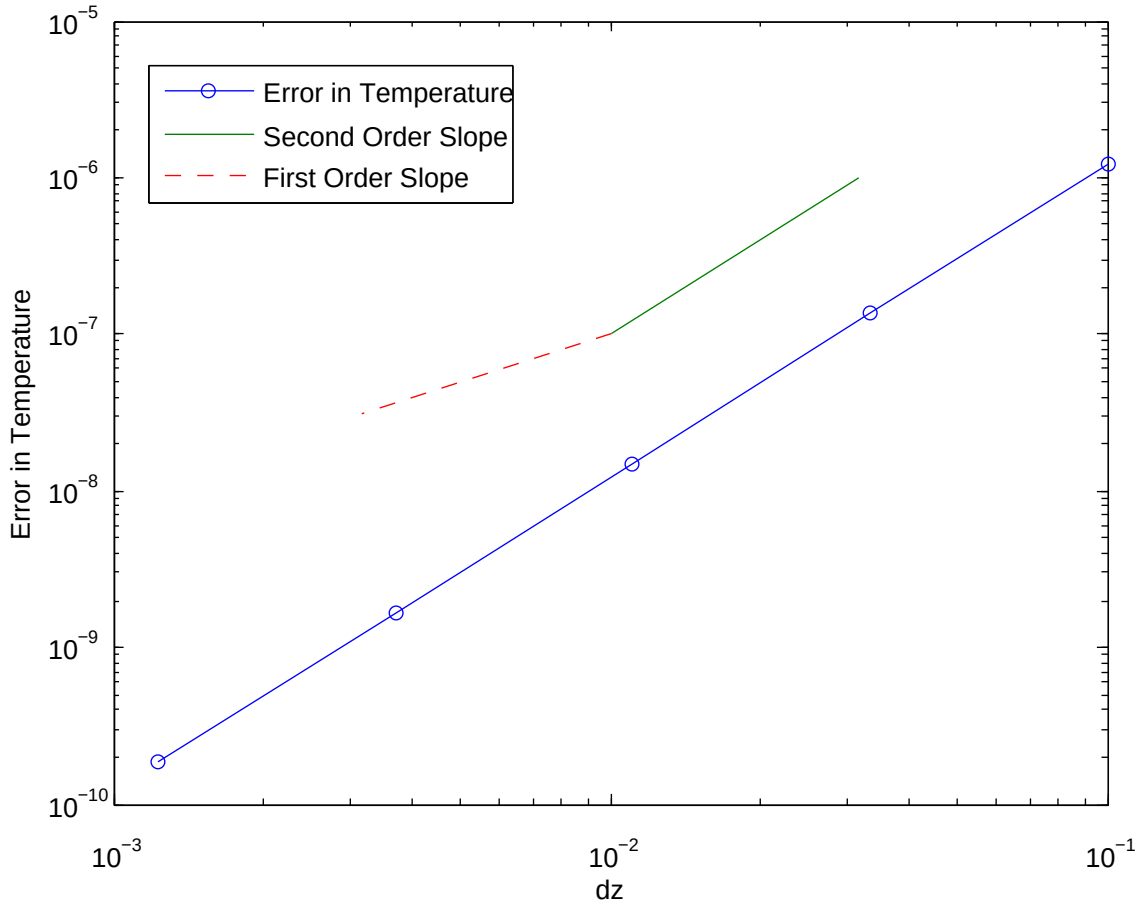

Figure 4.5: A log plot that confirms the order of error for the energy equation with respect to spatial step size.

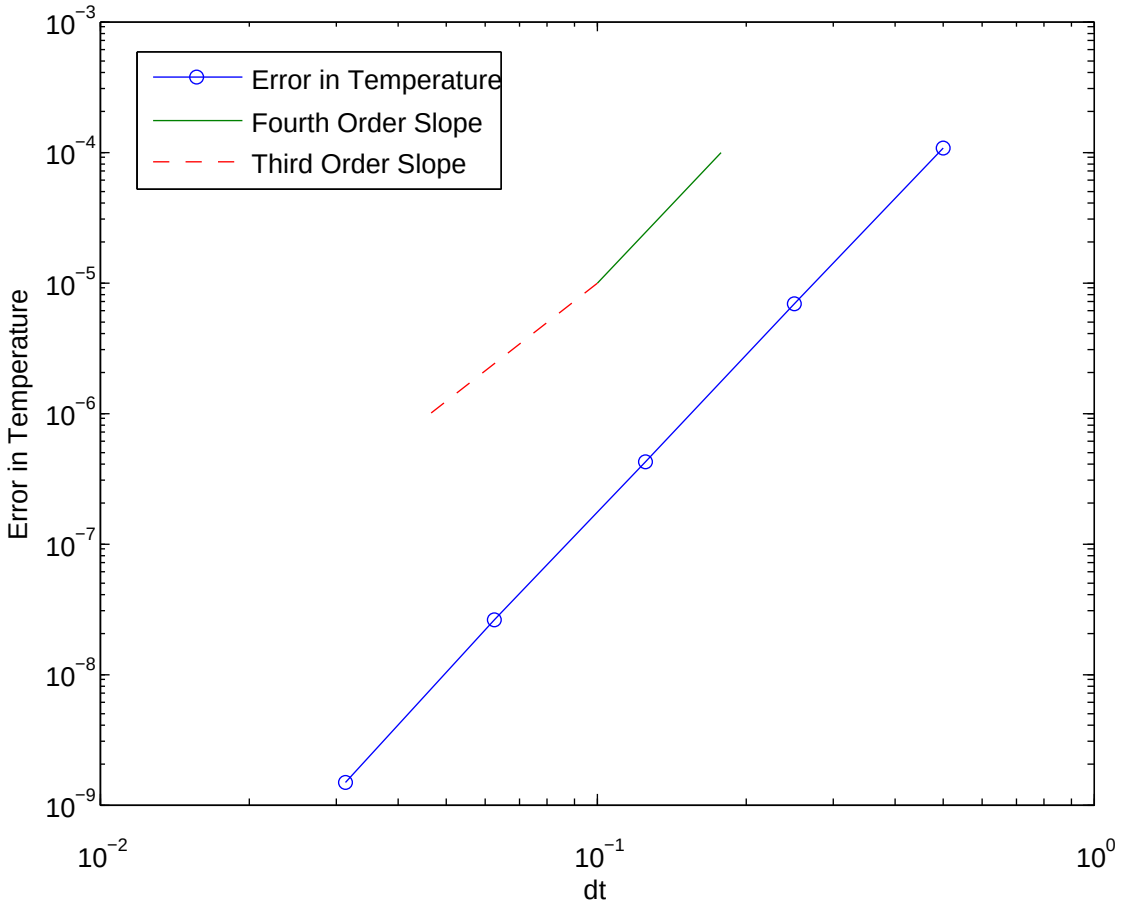

Figure 4.6: A log plot that confirms the order of error for the energy equation with respect to time step size.

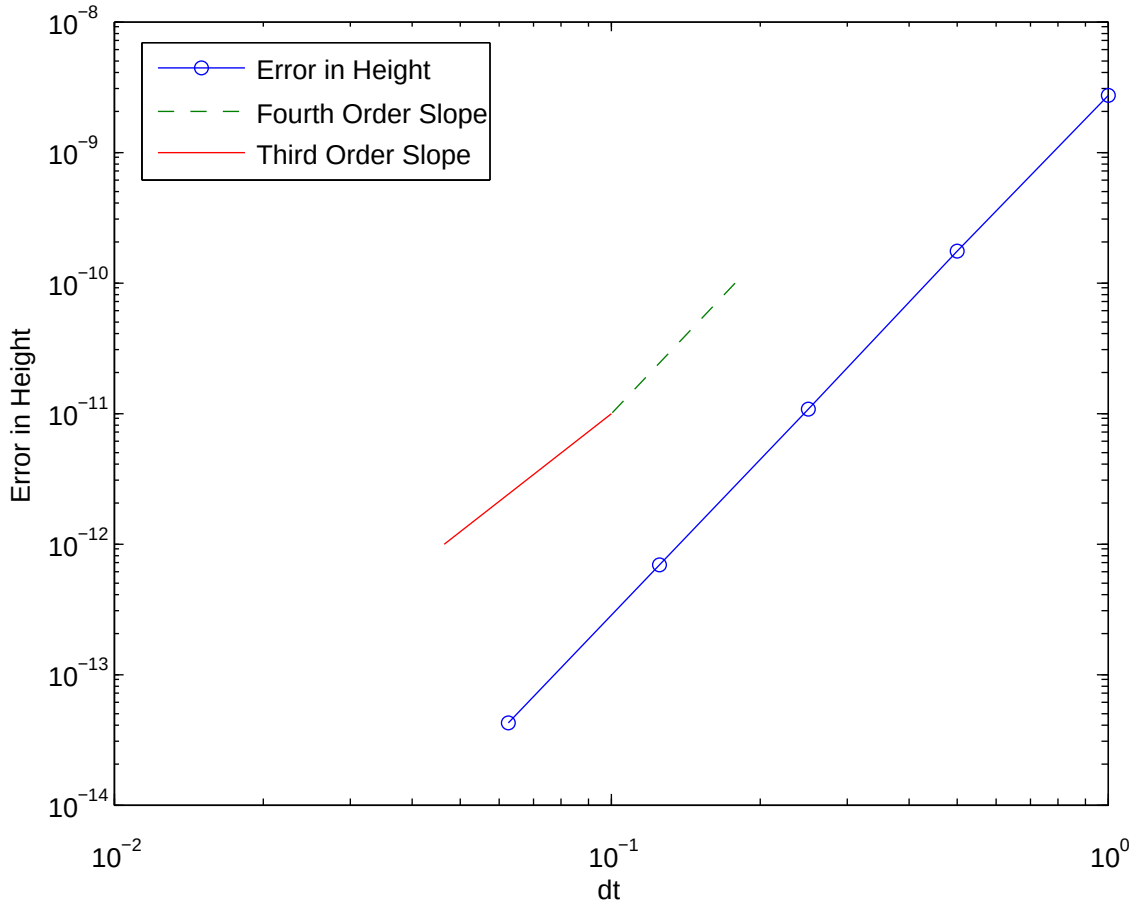

Figure 4.7: A log plot that confirms the order of error in the height equation with respect to time step size.

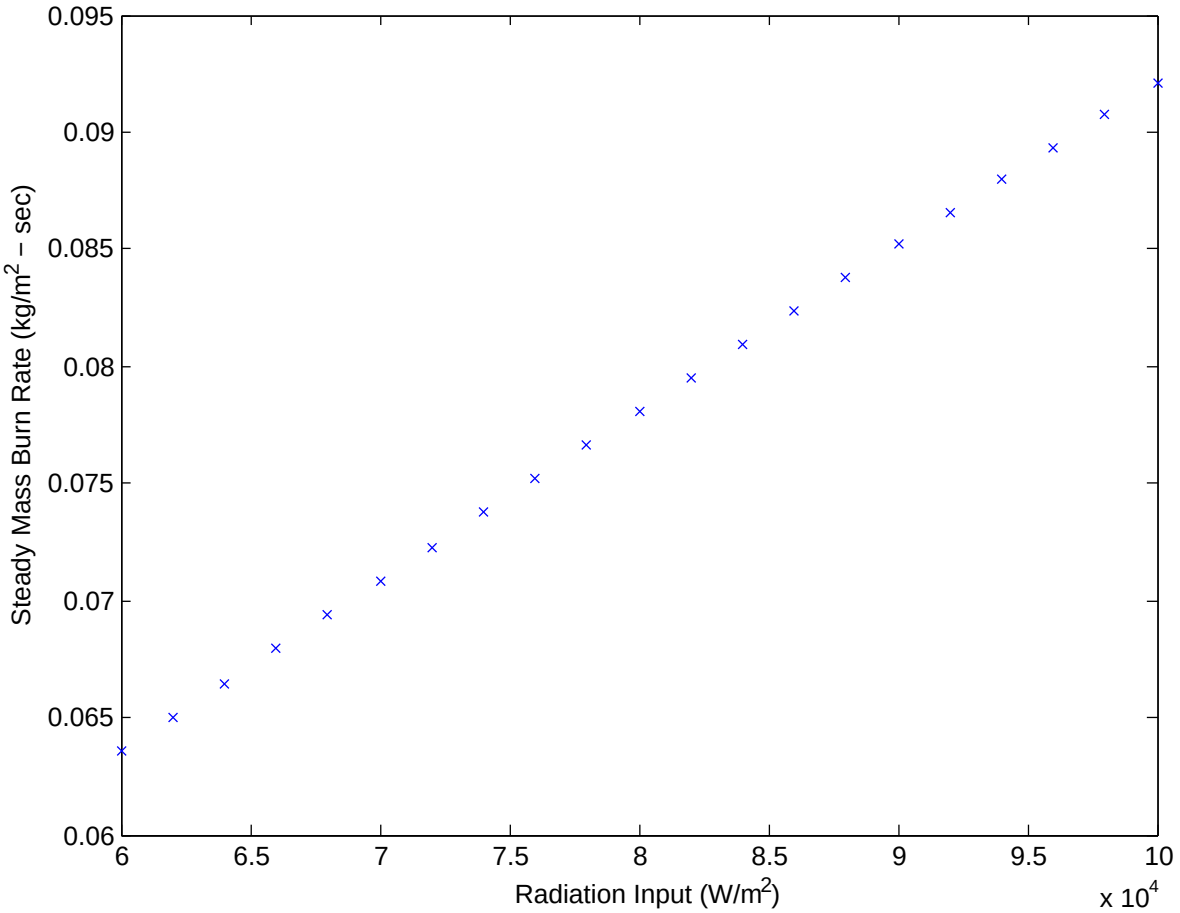

Figure 4.8: The response of the model output to radiative heat flux.

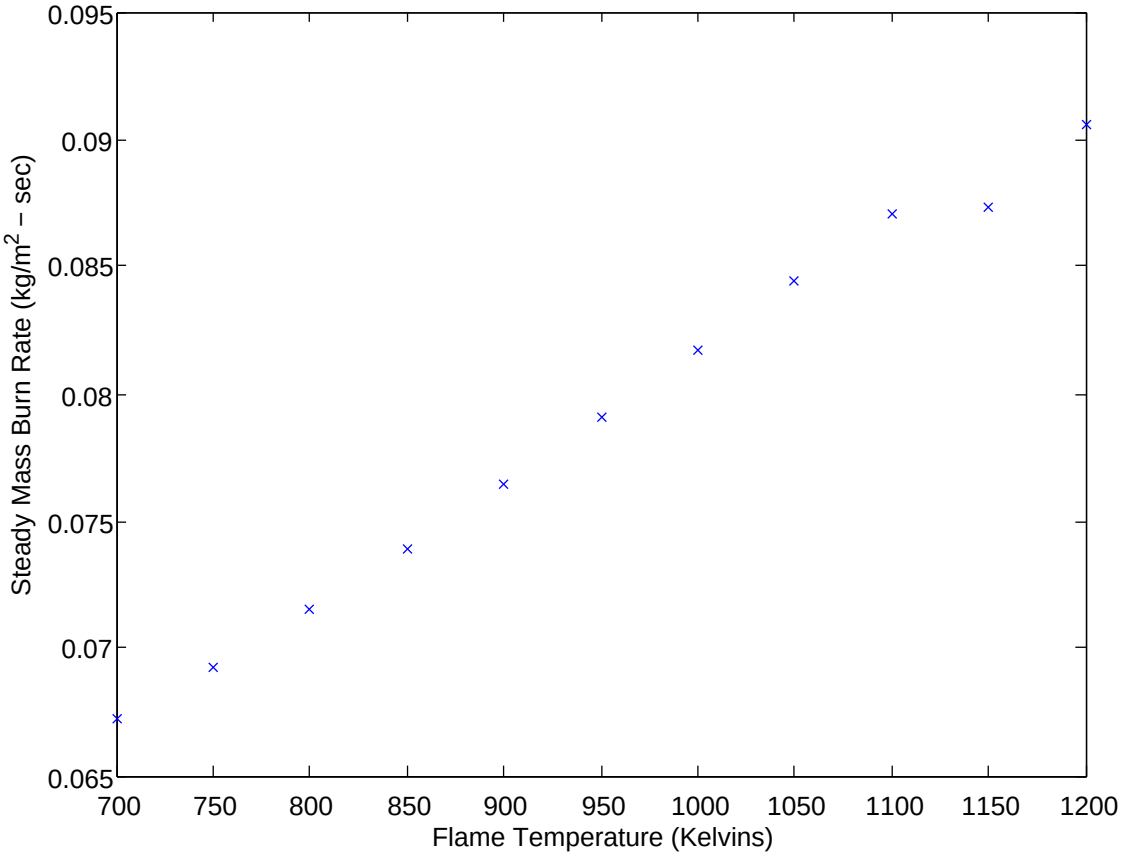

Figure 4.9: The response of the model output to flame temperature.

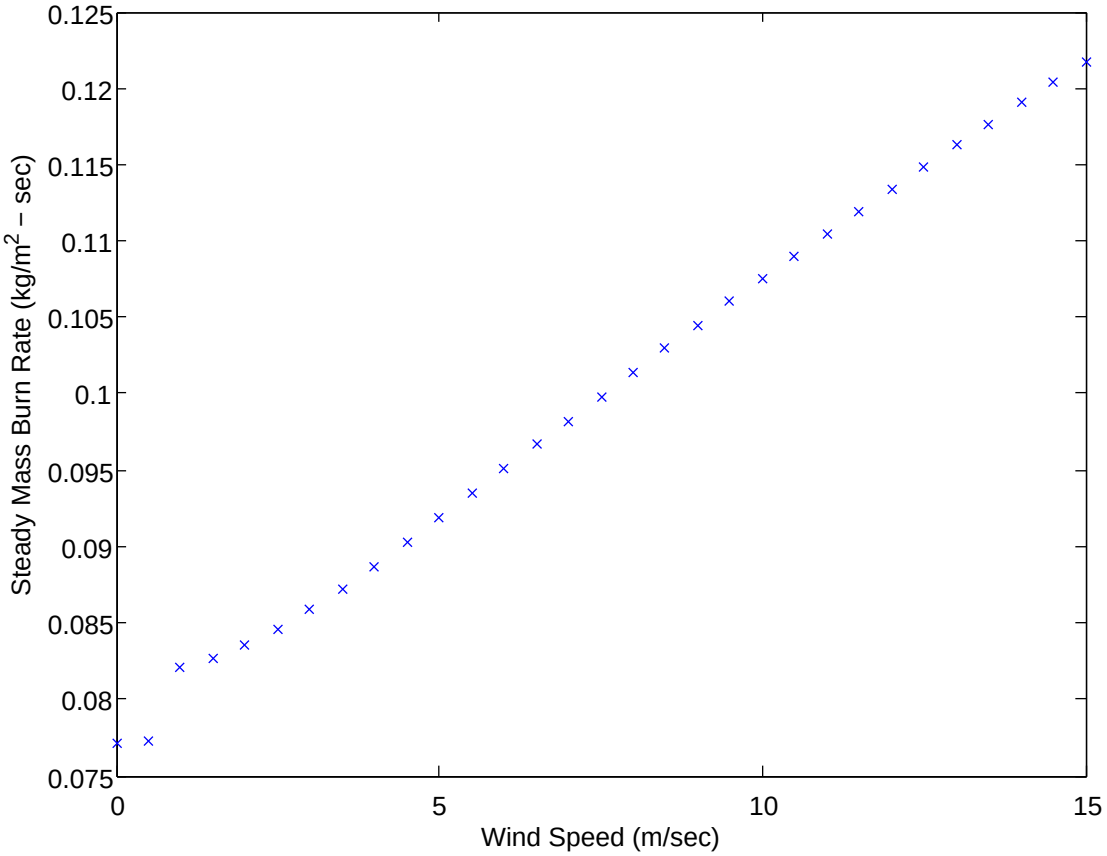

Figure 4.10: The response of the model output to variation in wind speed.

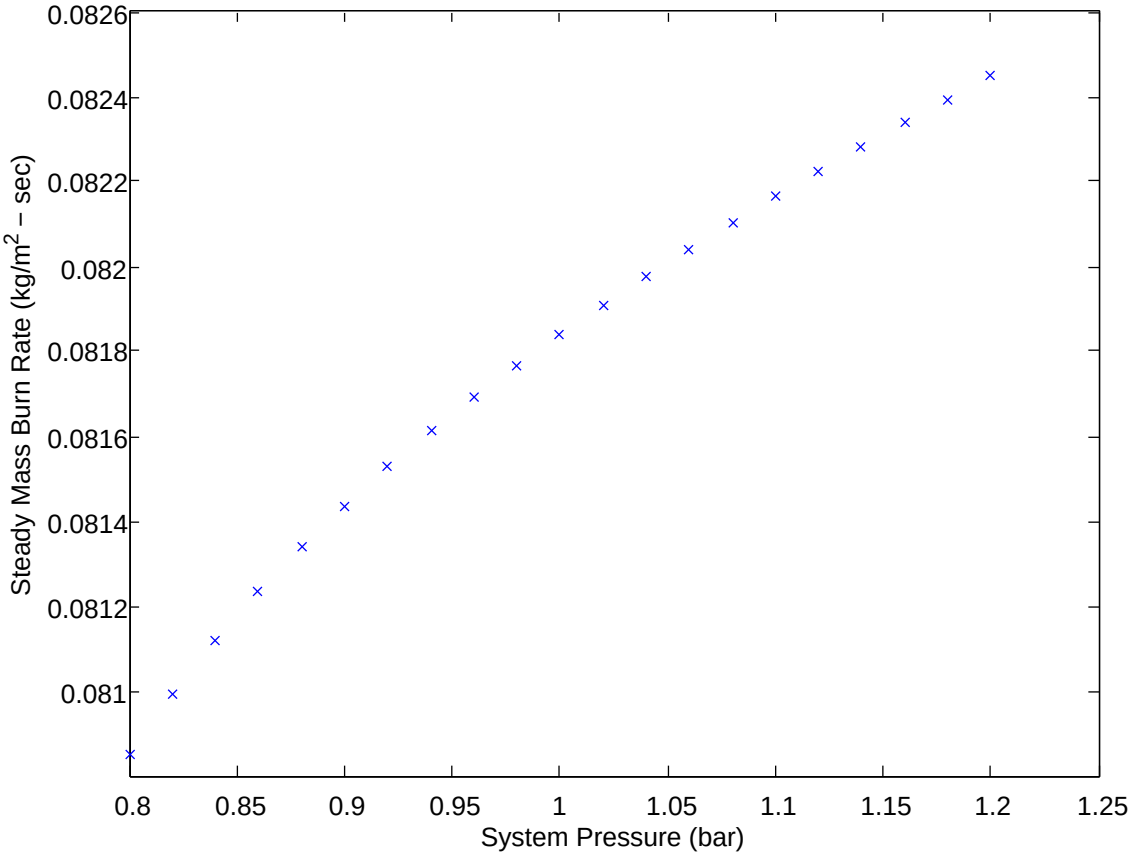

Figure 4.11: The response of the model output to variation in system pressure.

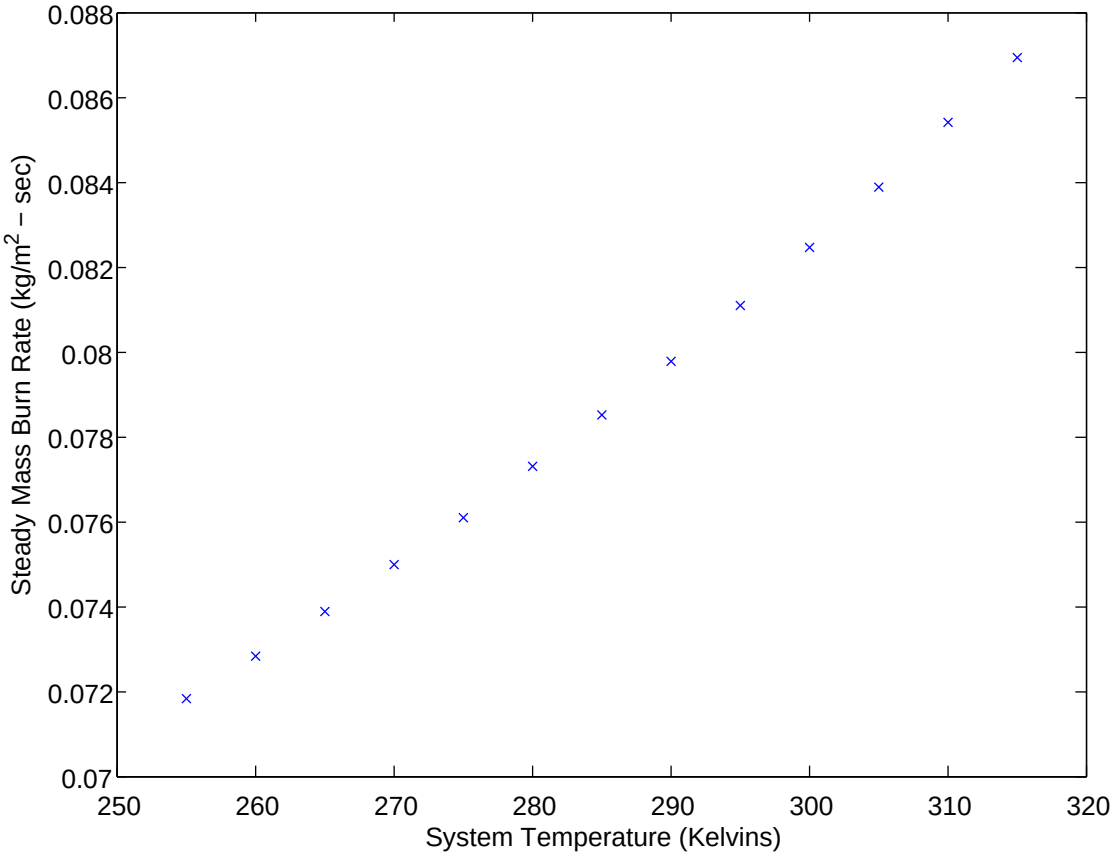

Figure 4.12: The response of the model output to variation in system temperature.

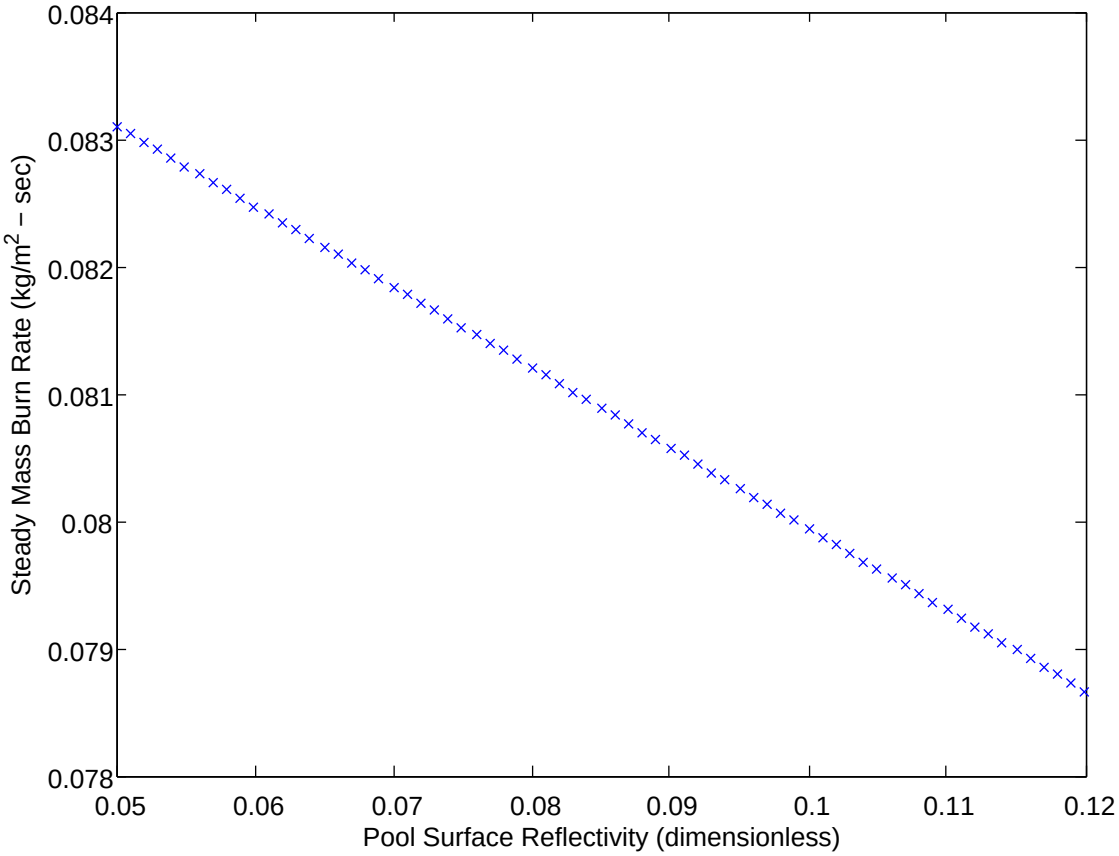

Figure 4.13: The response of the model output to variation in pool surface reflectivity.

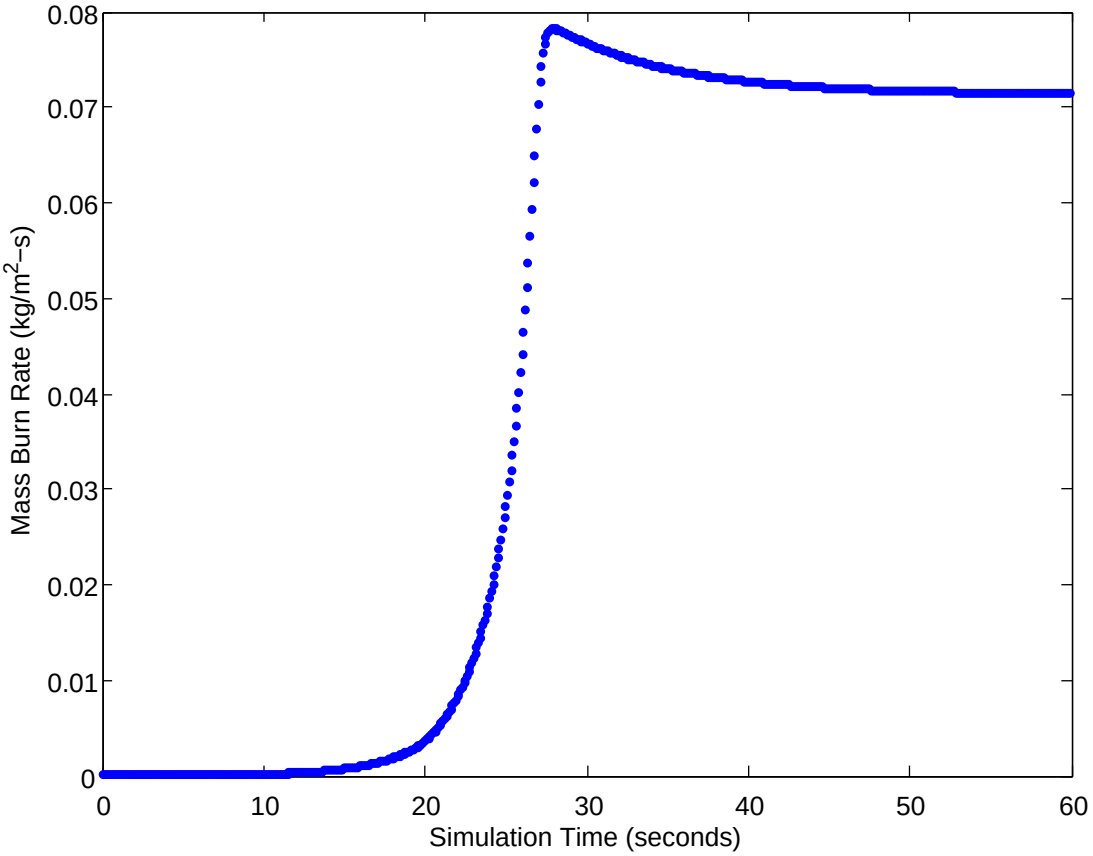

Figure 4.14: The simulated burn rate based on experimental conditions from case one from Blanchat et al. (2006).

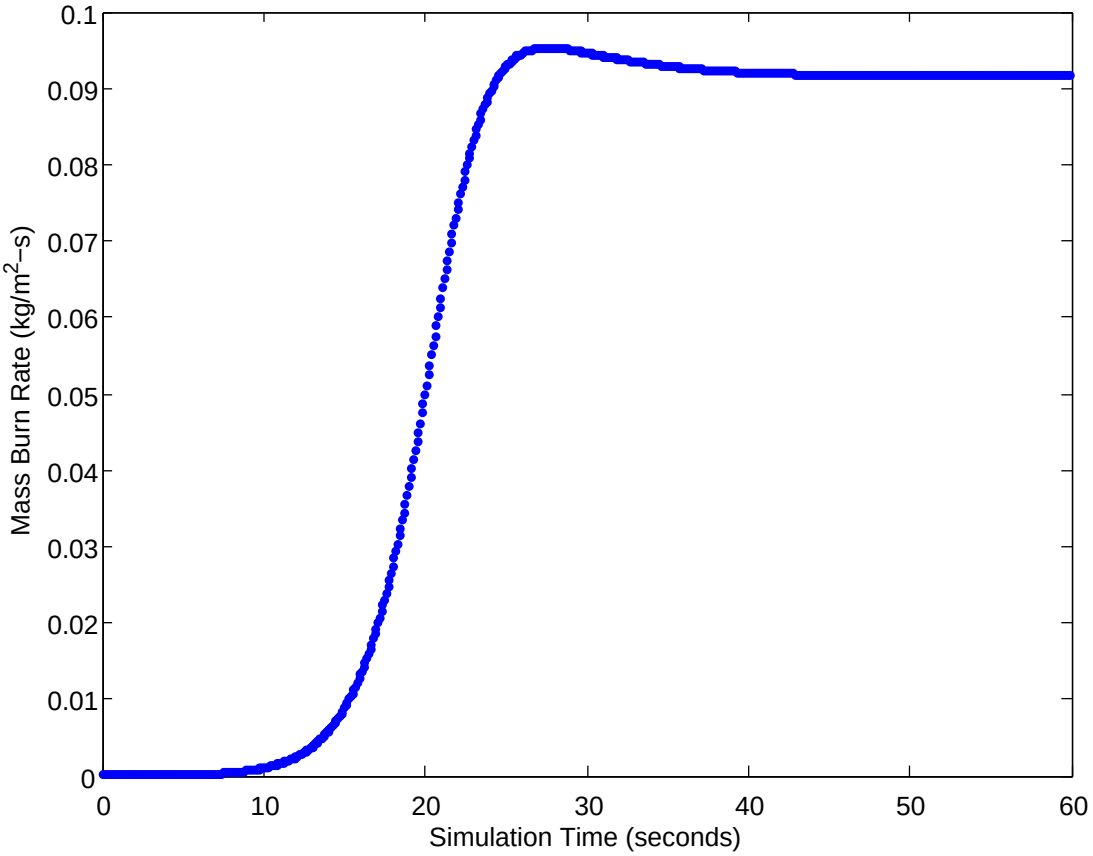

Figure 4.15: The simulated burn rate based on experimental conditions for case two from Blanchat et al. (2006).

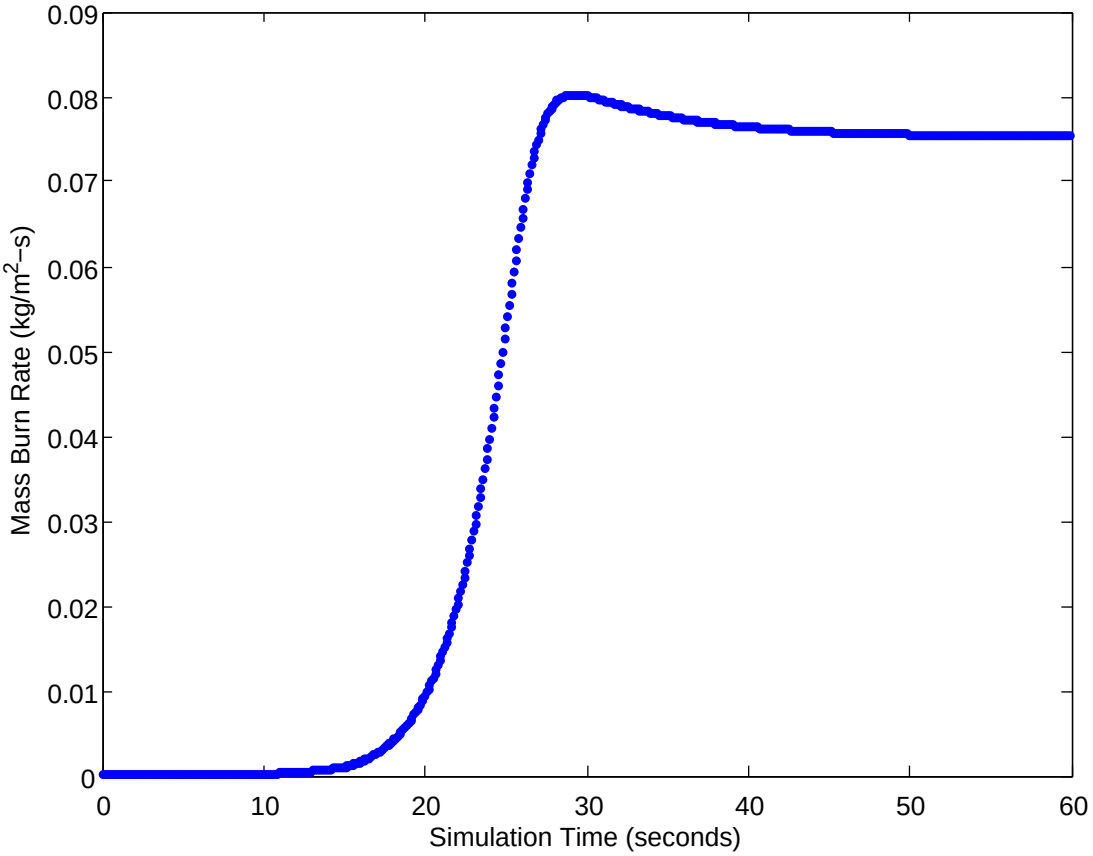

Figure 4.16: The simulated burn rate based on experimental conditions for case three from Blanchat et al. (2006).

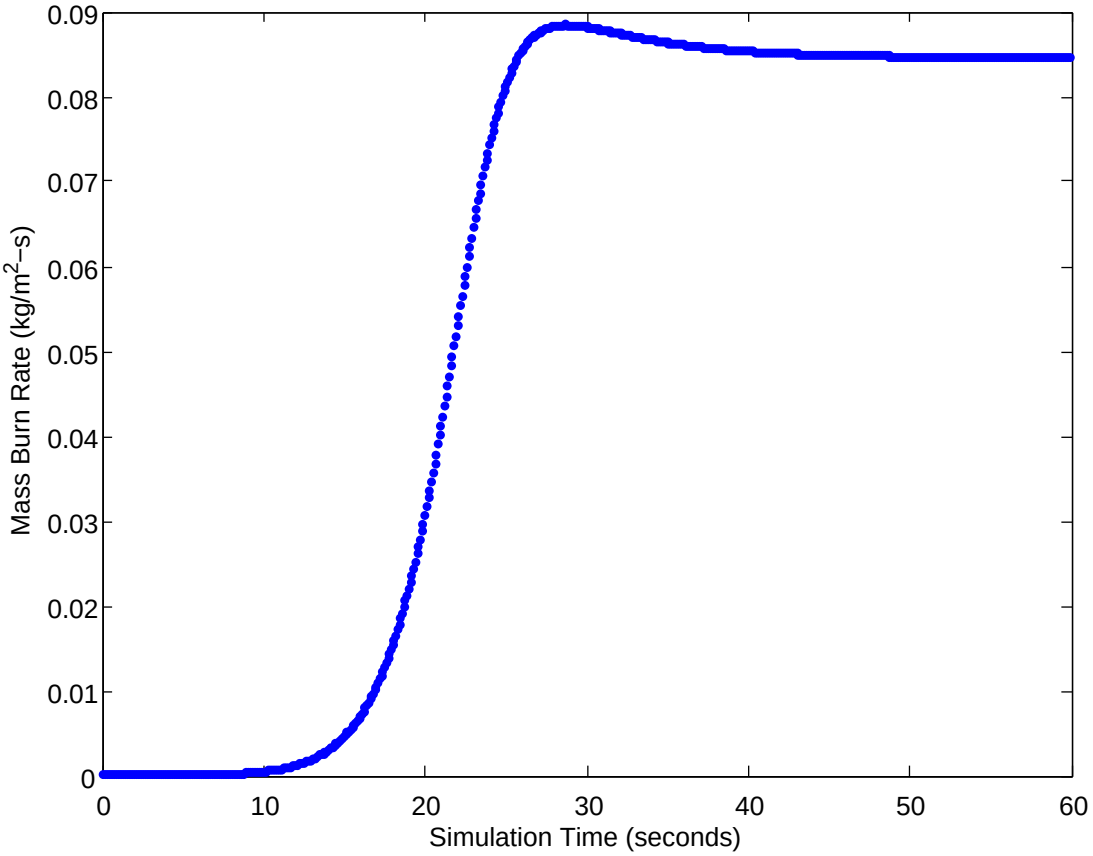

Figure 4.17: The simulated burn rate based on experimental conditions for case four from Blanchat et al. (2006).

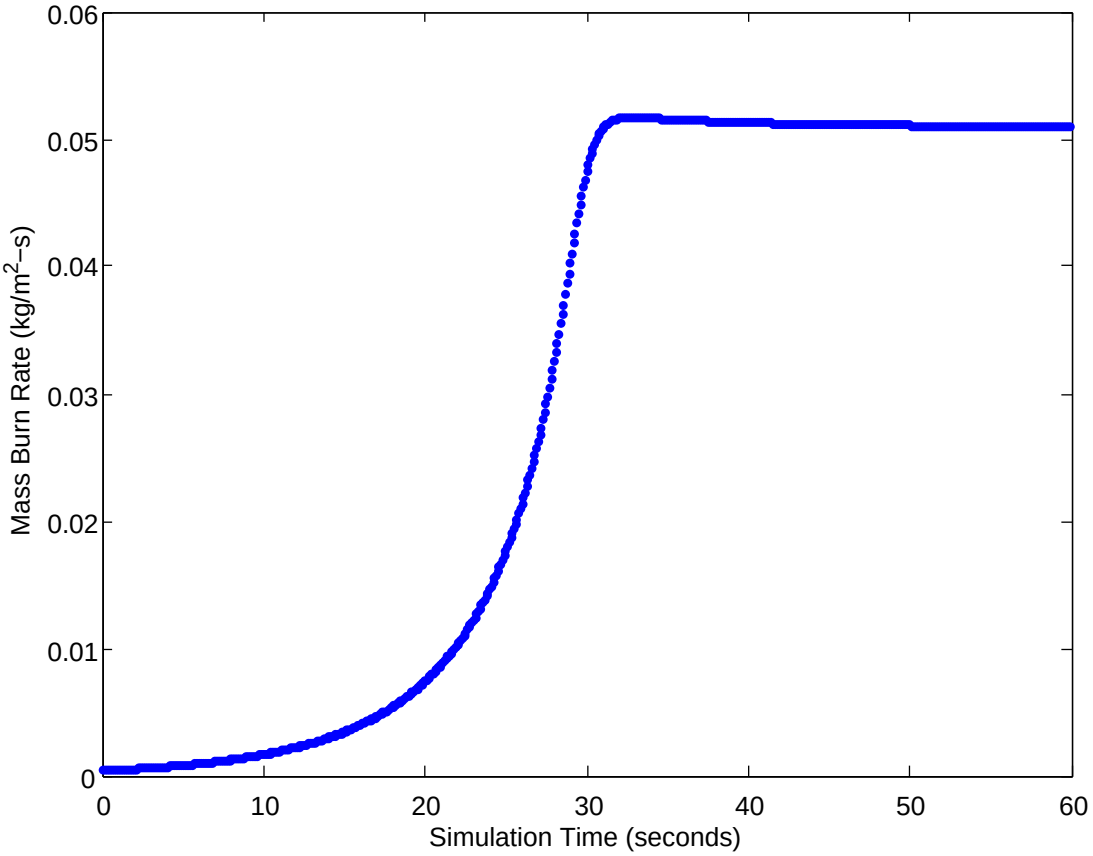

Figure 4.18: The simulated burn rate based on experimental conditions for the 30-centimeter heptane case from Klassen and Gore (1992).

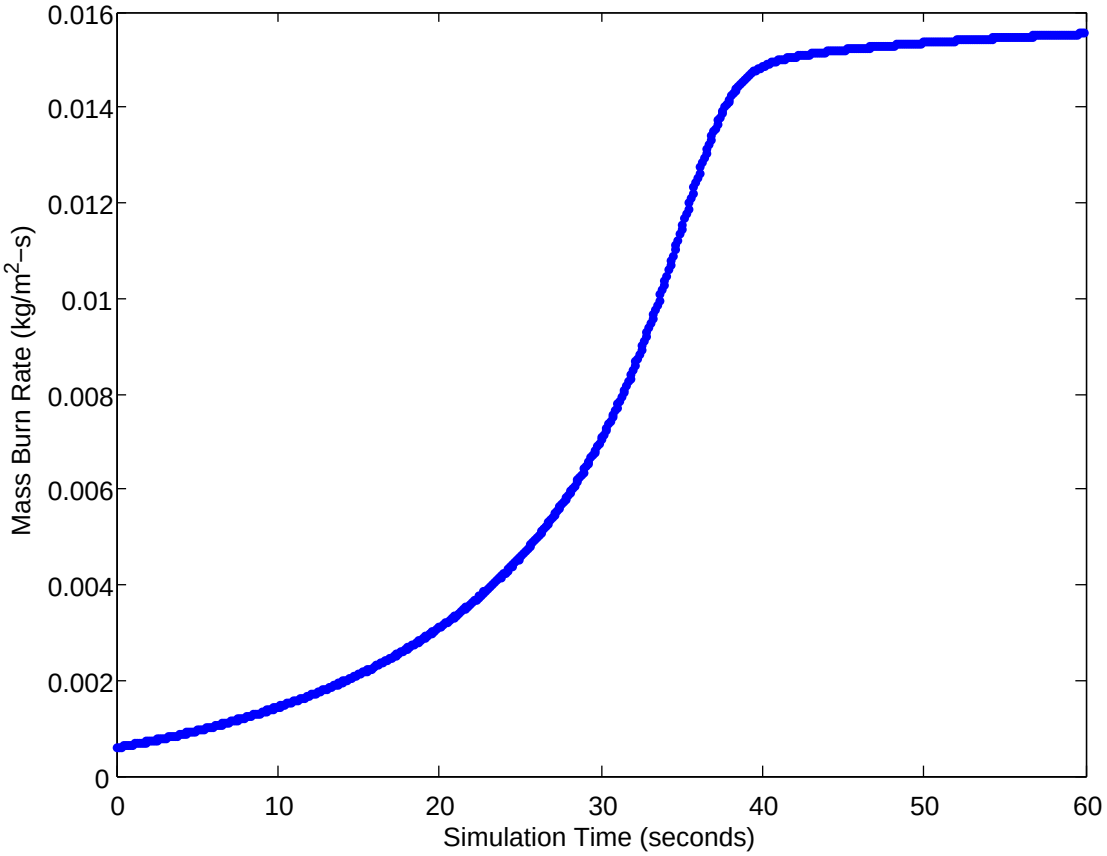

Figure 4.19: The simulated burn rate based on experimental conditions for the 30-centimeter methanol case from Klassen and Gore (1992).

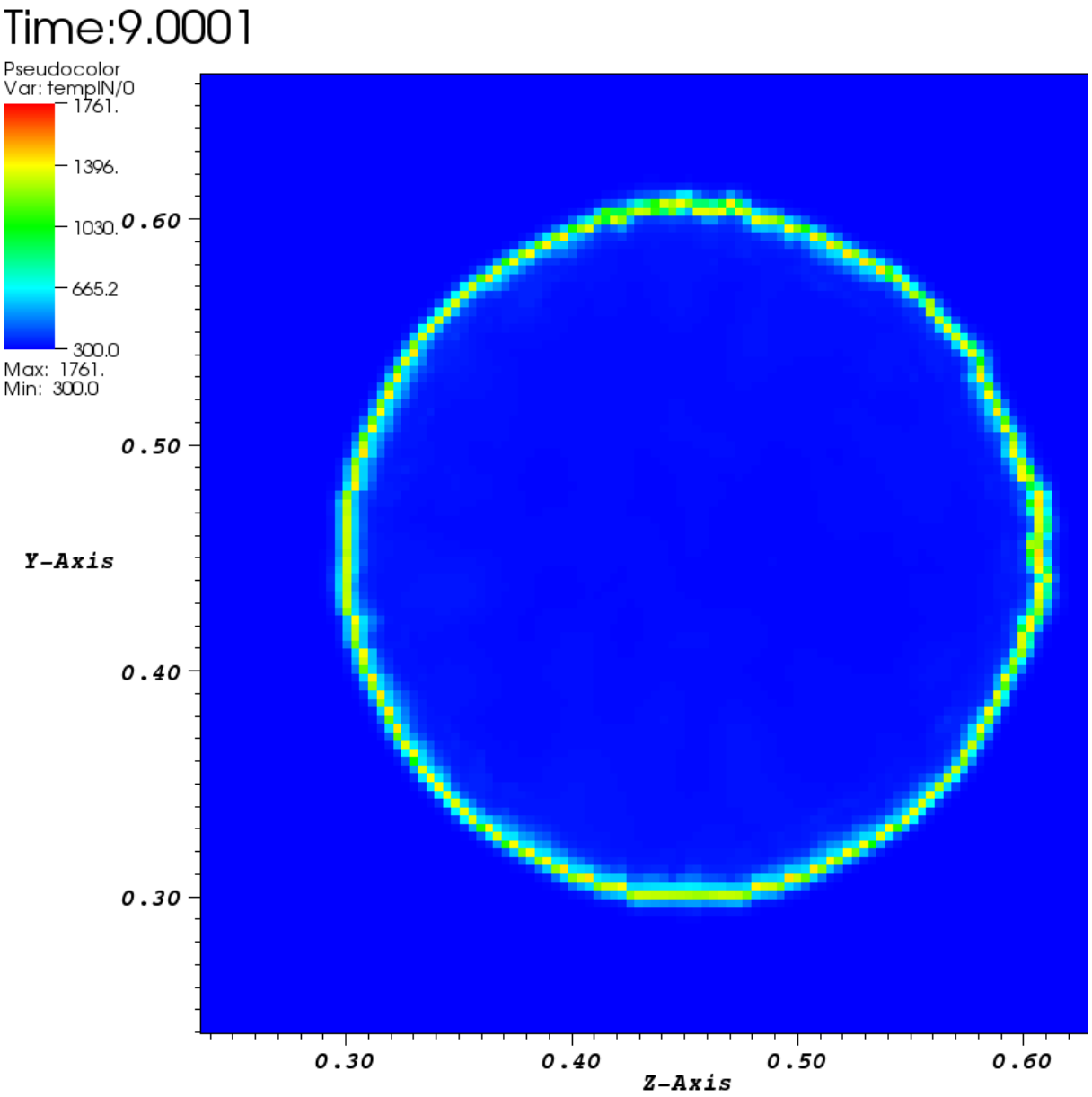

Figure 4.20: The flame temperature for a 30-centimeter heptane pool fire simulated in ARCHES.

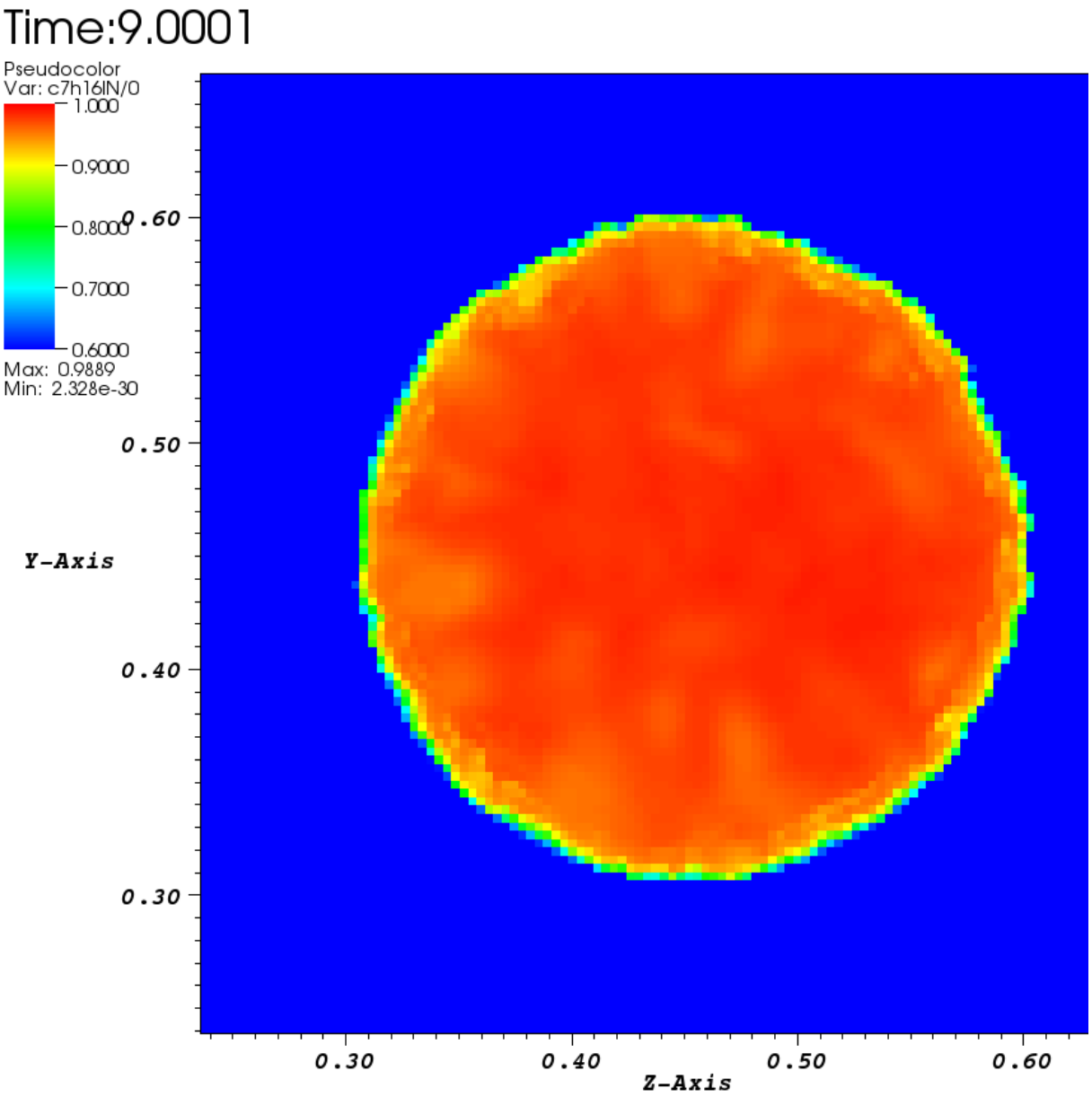

Figure 4.21: The heptane mass fraction above the pool for a 30-centimeter heptane pool fire simulated in ARCHES.

Table 4.1: The data for the grid convergence of liquid temperature with spatial step.

Case	**Δz (meters)**	**Temperature (K) (z = 0.34 meters)**	**Computed Order**	**Error Band (±)**	**Error Ratio (Target = 9)**
5	0.0400	298.157565			
4	0.0133	298.150843			
3	0.00444	298.150648	$p543 = 3.22$	$GCI3 = 2.43 \times 10^{-8}$	
2	0.00148	298.150628	$p432 = 2.09$	$GCI2 = 9.14 \times 10^{-9}$	2.688
1	0.000494	298.150626	$p321 = 2.04$	$GCI1 = 1.02 \times 10^{-9}$	8.997

Table 4.2: The data for the grid convergence of mass burn rate with spatial step size.

Case	**Δz (meters)**	**Mass Burn Rate ($\mathbf{kg/m^2 - sec}$)**	**Computed Order**	**Error Band (±)**	**Error Ratio (Target = 4.0)**
6	0.00187	0.074508			
5	0.000933	0.079397			
4	0.000467	0.082170	$p654 = 0.82$	$GCI4 = 5.531\%$	
3	0.000233	0.083234	$p543 = 1.38$	$GCI3 = 0.993\%$	5.57
2	0.000117	0.083516	$p432 = 1.92$	$GCI2 = 0.158\%$	6.28
1	0.000058	0.083586	$p321 = 2.00$	$GCI1 = 0.035\%$	4.51

Table 4.3: The data for the grid convergence for mass burn rate with time step size.

Case	**Δt (seconds)**	**Mass Burn Rate ($\mathbf{kg/m^2 - sec}$)**	**Computed Order**	**Error Band (±)**	**Error Ratio (Target = 2.0)**
6	0.500000	0.0633			
5	0.250000	0.0644			
4	0.125000	0.0678	$p654 = 1.11$	$GCI4 = 2.292\%$	
3	0.062500	0.0685	$p543 = 1.03$	$GCI3 = 1.236\%$	1.85
2	0.031250	0.0689	$p432 = 1.01$	$GCI2 = 0.622\%$	1.99
1	0.015625	0.0690	$p321 = 1.01$	$GCI1 = 0.312\%$	1.99

Table 4.4: The grid convergence data for the liquid height with time step size.

Case	**Δt (seconds)**	**Liquid Height (meters)**	**Computed Order**	**Error Band (±)**	**Error Ratio (Target = 2.0)**
6	0.500000	0.4996630			
5	0.250000	0.4996661			
4	0.125000	0.4996674	$p654 = 1.25$	$GCI4 = 2.38 \times 10^{-6}$	
3	0.062500	0.4996680	$p543 = 1.10$	$GCI3 = 1.35 \times 10^{-6}$	1.77
2	0.031250	0.4996683	$p432 = 1.05$	$GCI2 = 6.98 \times 10^{-7}$	1.93
1	0.015625	0.4996685	$p321 = 1.02$	$GCI1 = 3.55 \times 10^{-7}$	1.97

Table 4.5: Data for the grid convergence of mass burn rate without interpolation.

Case	Δt (seconds)	Mass Burn Rate ($kg/m^2 - sec$)	Computed Order	Error Band ($\pm$)	Error Ratio (Target = 16.0)
6	1.000000	0.0005625921			
5	0.500000	0.0005625922			
4	0.250000	0.0005625923	$p654 = 4.10$	$GCI4 = 1.06 \times 10^{-9}$	
3	0.125000	0.0005625923	$p543 = 4.05$	$GCI3 = 6.64 \times 10^{-11}$	15.95
2	0.062500	0.0005625923	$p432 = 4.02$	$GCI2 = 4.16 \times 10^{-12}$	15.97
1	0.031250	0.0005625923	$p321 = 4.02$	$GCI1 = 2.56 \times 10^{-13}$	16.25

Table 4.6: Test conditions and results for the simple sensitivity analysis.

Variables Tested	Nominal Input Value	Range Tested	Range of Output Nominal = 0.0818 $kg/m^2 - sec$
Radiative Input (W/m^2)	85,100	60,000 - 100,000	0.0636 - 0.0921
Flame Temperature (K)	1,000	700 - 1,200	0.0672 - 0.0905
Wind Speed (m/sec)	0.85	0 - 15	0.0772 - 0.1217
System Pressure (bar)	1	0.8 - 1.2	0.0809 - 0.0824
System Temperature (K)	298.15	255 - 315	0.0718 - 0.0869
Pool Surface Reflectivity	0.07	0.05 - 0.12	0.0831 - 0.0787

Table 4.7: Simulation input data taken from experimental pool fires.

Case	Fuel	Pool Diameter (meters)	Wind Speed (m/sec)	Average Radiation (W/m^2)	Steady Mass Burn Rate ($kg/m^2 - sec$)
Blanchat et al. 1	JP-8	7.93	0.85	85,100	0.0580
Blanchat et al. 2	JP-8	7.93	5.76	97,800	0.0710
Blanchat et al. 3	JP-8	7.93	1.66	88,700	0.0620
Blanchat et al. 4	JP-8	7.93	3.83	94,700	0.0670
Klassen/Gore Heptane	Heptane	0.30	0	18,000	0.0362/0.0331
Klassen/Gore Methanol	Methanol	0.30	0	10,000	0.0130/0.0115

Table 4.8: A comparison of simulation and experimental results.

Case	Steady Mass Burn Rate ($kg/m^2 - sec$)	Predicted Burn Rate ($kg/m^2 - sec$)	Discrepancy
Blanchat et al. 1	0.0580	0.0715	+23.3%
Blanchat et al. 2	0.0710	0.0916	+29.0%
Blanchat et al. 3	0.0620	0.0754	+21.6%
Blanchat et al. 4	0.0670	0.0847	+26.4%
Klassen/Gore Heptane	0.0362/0.0331	0.0510	+40.9%/+54.0%
Klassen/Gore Methanol	0.0130/0.0115	0.0155	+19.2%/+34.8%

CHAPTER 5

DATA COLLABORATION: SMALL HELIUM PLUMES

5.1 Introduction

The research reported in this chapter concerns validation activities with ARCHES simulations of laminar helium plumes and experimental observations made with holograms. This set of collaboration exercises concerns the lower right box in the validation hierarchy for pool fires depicted in Figure 1.2. Marcotte (2010) examines the holographic data generated by Schönbucher et al (1986), and provides data measurements that serve as the point of comparison for both the experimental and computational analyses. The quantity of interest and of comparison is the interference order, and this quantity is used as the simulation output. The helium plume is simulated using ARCHES. As ARCHES is a code designed for LES simulation, it is necessary to make adjustments to its code in order to simulate a laminar plume. Several plume simulations are run to test the range of uncertainty in the model input parameters. Three parameters are chosen as the most likely to influence the output of the simulation. The parameters are the helium inlet flow rate, the air co-flow rate, and the system temperature. Given 22 data point measurements of the interference fringe order it is initially determined from the consistency analysis that the whole dataset is inconsistent. Sensitivity analysis reveals that the upper uncertainty bounds of four data points have a dominant influence on the consistency measure. After re-evaluation of the error bounds for the dataset, a subsequent consistency analysis shows the dataset to be consistent.

5.2 Holographic Interferometry

Figure 5.1 is example of a hologram produced by Schönbucher et al. (1986) via holographic interferometry. The hologram shown in Figure 5.1 is an image taken from a

4.6-centimeter diameter helium plume. Figure 5.2 shows how lasers are utilized to produce holograms of buoyant plumes.

In holographic interferometry as applied by Schönbucher et al. a laser is produced with a wavelength of 415.15 nanometers (green light). The beam is split into two beams, and each beam is then expanded. One of the beams passes through air (reference beam), and the other passes through the plume. The two beams are re-combined and directed onto a holographic plate where the image is recorded. Because one laser beam passes through the buoyant plume, the volume through which it passes experiences variations in refractive index due to compositional variation in the helium plume. As the laser passes though the medium of variable refractive index its speed experiences slight changes. When the beam combines with the reference beam the two will be out of phase with each other, and their waves will interfere. The interference patterns are evident in Figure 5.1. Wherever the beams interfere constructively the hologram shows a brighter green color. Wherever the beams interfere destructively dark fringe lines appear. Holograms produced from holographic interferometry can provide information about certain plume properties. Each dark fringe line can be thought of in the same manner as contour lines on a topographical map. In the case of the map, the lines indicate changes in elevation. The more densely packed the lines are the steeper the change in elevation. With the hologram the dark fringe lines serve to represent changes in the refractive index of the plume. Wherever the fringe lines are found in higher density indicates a region of sharp variation in refractive index. Changes in refractive index in buoyant plumes reflect changes in density due to compositional variation and temperature variation. For helium plumes, the changes in refractive index can be entirely attributed to variations in the composition due to the diffusion of helium in air as there are no chemical reactions and the system is isothermal. Examination of the hologram depicted in Figure 5.1 shows that the largest concentration of fringe lines occur at the bottom of the image near the helium inlet. This indicates that the largest concentration gradients of helium occur at this location as pure helium first emerges from the inlet port and begins to mix with the air. As the helium moves away from the inlet the rate of mixing by diffusion decreases as concentration gradients diminish, and the fringe lines are spaced further apart.

Another important feature of holographic interferometry is that the images provide two-dimensional information for a three-dimensional system. The fringe lines found in holograms like that depicted in Figure 5.1 are the result of variations in refractive index over a path that runs orthogonal to the plane of the hologram. This reduction of information from a three-dimensional domain to a two-dimensional image is seen in the mathematical formulations for fringe order which will be described shortly.

The experimental data collected from Schönbucher et al. (1986) consists of videos of numerous helium plume experiments. Each frame in a video is a holographic image similar to that shown in Figure 5.1. The images are taken at rate of 1000 frames per second. Marcotte (2010) takes the holograms and adjusted the images to produce consistency in the brightness of the images. The images are then digitized and further processed in order to determine the time-averaged fringe order at different spatial locations over the inlet of the helium plume. The end result of the process is that fringe order values are measured with uncertainty for 11 radial locations at a height of 3 centimeters over the inlet and for 11 radial locations at a height of 5 centimeters over the inlet.

5.3 Important Mathematical Relations

As already stated, the chosen quantity of interest where experiment and computation meet is the interference fringe order. The justification for this choice arises from the amount of uncertainty involved in the experimental results if the fringe order is converted to other physical properties that can be compared more directly with ARCHES output. The processes needed to acquire fringe order data at different location already inflicts a degree of error in the measurements, and the mathematical steps needed to convert from the interference fringe order to other properties tend to magnify the error already present. While ARCHES does not directly calculate the interference fringe order for helium plume simulations, it does generate physical properties that can be used to obtain the desired output via postprocessing steps with simulation data. The interference order is obtained by integrating the adjusted refractive index along the path of the laser through the domain of the helium plume. Interference order is expressed:

$$S(x,y,t) = \frac{1}{\lambda}\int \left[n_M(x,y,z,t) - n_O\right] dz. \tag{5.1}$$

In this relation, S is the interference order, λ is the wavelength of the laser, n_M is the refractive index as a function of space and time, and n_O is a reference refractive index. The interference order is nonzero for domains where refractive index deviates from the reference value. For the helium plume system the value of the reference refractive index is that of pure air. Wherever there is helium in the system the refractive index will deviate from the reference value. The values of the deviation from reference will be zero where there is pure air, and in these locations there is no contribution to the integral that defines the interference order. As helium has a lower refractive index than air, where there is helium the deviation in refractive index from the reference will be negative. Therefore, in the regions of the domain where there is helium present the value of the interference order is negative. The values of interference order that correspond to the dark fringe lines seen in Figure 5.1 start at -0.5 and decrease by unity for each subsequent fringe line. In other words, fringes lines on the holograms occur for interference orders of -0.5, -1.5, -2.5, -3.5, -4.5, etc. In Figure 5.1 the outermost fringe corresponds to an interference order of -0.5, and the order decreases by unity with each adjacent fringe moving inward to the center of the plume.

As is evident from Equation 5.1, the interference order depends upon refractive index. Unfortunately, ARCHES does not compute refractive index directly. Another step in the postprocessing of ARCHES data must be included. Computation of the refractive index involves the application of the Gladstone-Dale relation, which can be expressed:

$$n_M(x,y,z,t) = \frac{3}{2}\left[\rho(x,y,z,t)\right]\sum_i \gamma_i(x,y,z,t) N_{O,\,i} + 1. \tag{5.2}$$

The refractive index depends on the density of the fluid, ρ, the composition which in Equation 5.2 is expressed as γ_i the mass fraction of species i, and the specific refractivity of species i, $N_{O,\,i}$. Gardiner et al. (1981) compute the values of the molar refractivties for several species including those for oxygen, nitrogen, and helium. Conversion to refractivity on a mass basis is a trivial conversion involving the molar mass of a species of interest.

In order to calculate the interference order for a helium plume simulation at any time, all that is needed from ARCHES is density and composition data for the whole domain at any time. ARCHES tracks the composition through a variable called the mixture fraction which for the helium simulations is the mass fraction of helium. The Gladstone-Dale equation can then be applied to obtain the refractive index at every point in the domain. Equation 5.1 is then applied, and the refractive index data are numerically integrated to reduce the three-dimensional domain to a two-dimensional set of interference order data. An example of a simulated hologram from a helium plume simulation in ARCHES is provided in Figure 5.3.

The contour lines in Figure 5.3 correspond to locations where the value of the interference order is consistent with values where destructive interference occurs. The lines in Figure 5.3 start at an interference order of -0.5 for the outermost line and increase in order to as low as -19.5 for the lowest line near the bottom of the domain near the inlet. Just as with the experimentally generated fringe pattern in Figure 5.1, the lines in Figure 5.3 are most densely packed together near the helium inlet and space themselves further apart at the higher regions of the plume. The fringe line pattern seen in Figure 5.3 is only for one time in the helium plume simulation. The fringe order data from experiment reflect values of interference order averaged over enough time to allow the passing of ten puffs of helium. In order to compare interference order from computation with that from experiment the computed orders are averaged over a similar time period. In this case an averaging interval about 2 seconds is required, and the values must be interpolated to the same locations where the experimental data are measured. Thus ARCHES with some postprocessing computes 22 interference orders at locations corresponding to the locations where the same quantity is measured experimentally.

5.4 Helium Plume Simulations

As already mentioned, the experimentally determined interference orders also include estimates of the uncertainty in the measurements. In order to perform a consistency analysis with the ARCHES simulation results for the same quantities it is necessary to determine the range of uncertainty in the simulation's output given that there is uncertainty

in the simulation input. This requires the determination of the active variables or the input variables with the greatest influence on the model's output. Table 5.1 lists the three variables considered most likely to have the greatest effect on the value of the simulation output.

Table 5.1 includes the nominal values of each variable and the range of uncertainty assumed for each variable. The helium inlet velocity is important because it directly affects the flow patterns of the helium as it mixes with the ambient air. The nominal experimental value for the inlet velocity is $0.135 \frac{m}{sec}$, and an uncertainty of $\pm 10\%$ is assumed with this measurement. The system temperature has a potential to influence the output of the simulation because the diffusivity and the viscosity of helium and air are functions of temperature, and these properties affect the mixing properties of the helium plume with the air which in turn affects the interference order. The range of temperatures used in this series of simulations is based on weather data for Stuttgart, Germany at the time Schönbucher et al. (1986) perform the helium plume experiments. The air co-flow velocity is a phenomenon that is incorporated into the ARCHES simulations as a boundary condition. The air co-flow phenomenon arises from the fact that as the helium rises from the inlet it entrains the surrounding air inducing an air flow pattern that cannot be properly simulated without either expanding the computational domain at a tremendous computational cost or by adding an air inlet boundary condition at the bottom of the domain. The latter option is applied in the simulations. This phenomenon is especially important when the helium inlet is elevated from the ground as is the case with the experiments that Schönbucher et al. perform. The effect of air entrainment is very important to capture as the mixing of helium and air is what the interference order measures. There is less certainty as to what an appropriate air co-flow velocity would be, so a range of 10% to 50% of the nominal helium inlet velocity is assumed for this property for this series of simulations.

With the proposed active input variables chosen with their range of uncertainty, the next step is to run a series of simulations that test the range of input values to the simulation and see the resulting range of output. Since the data collaboration software cannot obtain output from ARCHES in any reasonable time, it is necessary to run a certain numbers of ARCHES cases each using different input values for the parameters shown in Table 5.1.

With the input and output data from the simulations a quadratic model is then formulated and applied to the collaboration software, and the consistency analysis can be performed. To obtain a sufficient amount of response data from ARCHES, a Box-Behnken design approach is applied to the ARCHES simulations. For the case of three input variables, 13 simulations are needed to obtain a sufficient number of data points to form a quadratic model. Table 5.2 shows the input values for the 13 simulations of the helium plume.

Examination of the different values of the inputs for each simulation shows that the values of the inputs correspond to the nominal, upper bound, and lower bound values shown in Table 5.1. Each simulation shown in Table 5.2 simply tests different combinations of the inputs to get a range of the simulation output. Once the output from each simulation is obtained, the data are then applied to form a quadratic model that gives the model's output (the interference order at the previously mentioned 22 spatial locations) as a function of the three input parameters. The form of the quadratic model is:

$$\begin{aligned} y(x_1, x_2, x_3) = b_0 \; + \; b_1 x_1 \; + \; b_2 x_2 \; + \; b_3 x_3 + \\ b_{12} x_1 x_2 \; + \; b_{13} x_1 x_3 \; + \; b_{23} x_2 x_3 \; + \\ b_{11} {x_1}^2 \; + \; b_{22} {x_2}^2 \; + \; b_{33} {x_3}^2. \end{aligned} \tag{5.3}$$

In equation 5.3, y is the model output as a function of the three varied inputs, x_1, x_2, and x_3. The different parameters b, are the quadratic fit parameters. The fit is obtained via a least squares approach using the output data from each simulation along with the input parameters. As there are 22 spatial locations where the interference order is tracked, there are 22 quadratic model fits of the form shown in Equation 5.3 corresponding to each data point.

Figure 5.4 is a contour plot of fluid density arising from an ARCHES helium simulation. This image illustrates the computational domain used to simulate the helium plume.

The computational domain for each helium simulation is a cubic volume 13.8 x 13.8 x 13.8 centimeters in dimension. A length of 13.8 centimeters is chosen because the helium inlet diameter is 4.6 centimeters, and the rule of thumb is to use a domain whose length is at least three times the diameter of the inlet. A smaller domain can lead to instabilities caused by the proximity of the side boundaries to the plume. The helium inlet is circular

in shape and placed at the center of the bottom of the domain. The velocity of the helium inlet is constant through the simulation, and it is one of the parameters varied over the set of simulations. The entire bottom square of the domain is treated as an air inlet whose velocity is also constant and is also one of the three variable parameter inputs to the helium simulation. The four sides of the simulation domain are pressure boundary conditions, and the top wall is an outlet boundary condition. Each simulation is run for a simulation time of about six seconds. This is enough time to allow the helium plume form and reach a steady behavior that can then be used for analysis and postprocessing.

One of the complexities in using ARCHES for laminar helium plumes is that ARCHES is designed for turbulent flows using LES. It is necessary to alter ARCHES to be able to handle laminar cases such as the small helium plumes. In turbulent flows, the mixing is dominated by turbulent effects as opposed to molecular diffusion. This is not the case with the laminar helium scenarios. In order to model the mixing of the air/helium mixture in the laminar case, the constant-coefficient Smagorinsky subgrid scale model is applied and the coefficient is set to zero. This eliminates the effects of turbulent mixing on the subgrid scale. Next, it is necessary to provide a laminar diffusivity to model the mixing effects due to molecular diffusion. A Schmidt number is calculated for each node in the domain based on the system temperature and the composition at each node. To obtain the Schmidt number, the fluid viscosity and the diffusivity of air/helium mixtures must be estimated for air/helium mixtures at different temperatures and compositions. The viscosities of air and helium are computed from a polynomial fit based on data provided by Incropera and Dewitt (2001), and the viscosity of the air/helium mixture was estimated using the method developed by Reichenberg (1977). The binary diffusivities for the air/helium system are estimated through application of the method given by Wilke and Lee (1955).

Each of the 13 simulations listed in Table 5.2 is run using a three-dimensional compuational domain that is partitioned into cube-shaped nodes and using a resolution of 100^3 nodes. Obtaining interference orders for the whole domain as shown in Figure 5.3 is time intensive and unnecessary for the purposes of the consistency analysis. All that is needed is the interference order at certain locations in the reduced two-dimensional domain where the interference order is defined. The measurements for these properties are taken at

heights of 3 centimeters and 5 centimeters above the inlet and at different radial locations over the inlet. The locations of the computational nodes do not necessarily coincide with the measurement locations; therefore, interpolation will need to be applied to obtain the computational data at the appropriate locations in the domain. In the postprocessing step of the computational data analysis it is not necessary to compute the interference order at all heights and radial locations. Instead, fluid density and mixture fraction are extracted from the parts of the domain near the heights of 3 centimeters and 5 centimeters. Because of the way the domain is divided there is no node that is positioned exactly at 3 centimeters or 5 centimeters above the inlet. Instead all the relevant property data are extracted from the two heights immediately above 3 centimeters and the two heights immediately below the 3-centimeter height. The same is done for the 5-centimeter height. This extraction gives four planes parallel to the helium inlet that are adjacent to the 3-centimeter plane and four planes adjacent to the 5-centimeter plane. The next step is to take the density and mixture fraction data defined at the planes just mentioned and apply the Gladstone-Dale relation (Equation 5.2) to obtain the refractive index at all nodes adjacent to the heights of interest. The refractive index at a location in the domain that is known to have a zero or near-zero mixture fraction (pure air) is taken and used as the reference refractive index in Equation 5.1. This equation is applied to every point in each of the planes of data to give the interference order at four lines adjacent to each of the heights of interest. Because the experimental data are time-averaged over several puffing cycles, the procedure just described for obtaining interference orders at different heights of interest in the domain is repeated for each computational time step (every 0.01 seconds) for a time interval of 2 seconds. The chosen time interval for each simulation is from 3.5 to 5.5 seconds. This gives interference order data at eight heights for about 20 computational times. The data from the 20 times are then averaged to give time-averaged interference orders at the eight heights: the four closest to the 3-centimeter height and the four closest to the 5-centimeter height. Values for the interference order at 3 centimeters and 5 centimeters of height for all available radial locations are computed using Lagrangian interpolation of the four adjacent lines. This reduces eight lines of output data to two lines: one line at 3 centimeters of height and one line at 5 centimeters. The last step is to interpolate on the radial locations

to obtain the output at the appropriate locations. This gives 22 data points from the simulation that can be compared with the experimental data.

The postprocessing procedure just described is repeated for each of the 13 simulations. The results for the base case at the 22 locations are used as the nominal values. An upper bound and lower bound for each of the 22 data points are also chosen depending on which simulation provides the highest and lowest prediction for each of the 22 data points. The outputs from the simulations for each of the data points with the uncertainty due to input uncertainty are plotted with the same quantities from experiment in Figures 5.5 and 5.6. Figure 5.5 shows the computational/experimental comparison for the data points taken at different radial locations at a height of 3 centimeters. Figure 5.6 shows the same for a height of 5 centimeters. In both figures a radial location of zero indicates the center of the helium inlet.

The uncertainty shown in Figures 5.5 and 5.6 for the computational data reflects only the uncertainty in the output due to uncertainty in the experimental conditions that serve as model inputs. Since the simulations are run with the turbulence models turned off it is not thought that error from such models will be of significance in these laminar cases. The densities of air and helium are assumed to conform to ideal gas behavior. The assumption has been verified against compressibility correlations given by Mansfield and Peck (1964) for helium, and Ciddor (1996) for air. The estimation methods for the diffusivites calculated for the air/helium system are reported by Poling et al. (2001) to have errors as high as 2% in the temperature range of interest. For nonpolar gases the error in gas mixture viscosities estimated from Reichenberg's method (1977) as reported by Poling et al. is less than 2%. These sources of error are unlikely to impact the model output as significantly as the input uncertainty.

Another possible source of error to the simulation output is due to spatial discretization. Temporal discretization is not considered important given the small time step sizes. To gauge the effect of spatial discretization error the original base-case simulation is taken and two other similar simulations are run at different spatial resolutions. The three simulations are similar in that they use the same input parameters as shown for the base-case in Table 5.2. The original simulation applies a mesh resolution of 100^3 nodes. One of the simulations

uses a coarser grid of 80^3 nodes, and the last simulation uses a finer grid of 125^3 nodes. The data from the coarser and finer resolution cases are processed in the same manner as for the data from the original simulations as already described earlier. The same 22 data points are compared on the same plots for the 3-centimeter and 5-centimeters heights for the three simulations, and are depicted in Figure 5.7 for the 3-centimeter height data and in Figure 5.8 for the 5-centimeter height data. Examination of the data points shows that there is negligible difference among the different mesh resolutions. The error arising from spatial discretization is likely insignificant compared to the uncertainty arising from the input parameters in the model.

5.5 Consistency Analysis: First Iteration

Twenty two data points along with their experimental uncertainty and their model predictions make 22 dataset units which form a dataset for consistency analysis. The dataset units are numbered 1 through 11 for the 11 points measured at 3 centimeters height. These points are shown in Figure 5.5 going in numerical order from left to right. The 11 points shown in Figure 5.6 are associated with dataset units 12 through 22 going in numerical order from left to right. Examination of the experiment/model comparison data in Figures 5.5 and 5.6 shows that there are four data points: 1, 12, 13, and 22 that give cause for concern. With these four dataset units there is no region of overlap for the uncertainty in the experimental data and the model predictions.

5.5.1 Pairwise Consistency Analysis

After visual inspection of the data in Figures 5.5 and 5.6 the next step in the consistency analysis is to perform pair-wise consistency analyses on each possible pair in the dataset. For dataset units 1 through 22, every possible pair is used to make a 2-member dataset, and a consistency analysis is performed on each pair. This procedure can reveal additional inconsistencies that may not be apparent from visual examination of the raw data. The results from the pair-wise analysis for all possible pairs are illustrated in Figures 5.9 and 5.10. The results are divided into two figures to keep the presentation of the data more clear. The data shown in Figures 5.9 and 5.10 are bar graphs of the consistency measure for each pair-wise analysis. A positive value on the graph indicates a pair of dataset units

which are consistent with each other. A negative value indicates an inconsistent pair. The first multi-colored bar in Figure 5.9 represents the consistency measures for all pair-wise analyses involving dataset unit 1. Within that bar are 22 smaller bars, each indicating the results of specific consistency analysis. The first small bar indicates the consistency measure for first dataset unit with itself. The second small bar is the same quantity for the pair-wise test with the first and second data set units, and so on. The last small bar in that cluster represents the results for the analysis of the first dataset unit paired with the last dataset unit. The next large, multi-colored bar in Figure 5.9 then represents consistency measure results for all pair-wise tests involved the second dataset unit, each smaller bar representing the result for each test in numerical order. Given the way this set of data is organized it is important to note that their is redundancy in the data depicted in Figures 5.9 and 5.10. For example, the first small bar in the last bar cluster of Figure 5.10 has the same value as the last small bar in the first bar cluster of Figure 5.9. Both of these bar indicate results for a pair-wise consistency analysis involving dataset unit 1 and dataset unit 22.

As previously mentioned from the examination of data shown in Figures 5.5 and 5.6, four data points (1, 12, 13, and 22) give cause for concern with achieving consistency in the dataset. Since these four points are inconsistent on their own, any pair-wise test involving these four points is expected to also be inconsistent. Examination of the pair-wise tests shown in Figures 5.9 and 5.10 does indeed show that any pair-wise test involving any of the problematic four points is inconsistent as is evident from the first bar cluster in Figure 5.9 and the first, second, and last bar clusters in Figure 5.10 as well as the first, 12^{th}, 13^{th}, and 22^{nd} bars in all bar clusters in both Figure 5.9 and Figure 5.10. Fortunately, any pair-wise test that is absent of any of these four points is shown to be consistent. This suggests that because of the inconsistency of the four points, the whole dataset (all 22 points) will be inconsistent, but the source of the inconsistency will largely arise from the four points.

5.5.2 Analysis of the Full Set

The next step in the consistency analysis is to perform the consistency test on the entire dataset. Since four dataset units have already been shown to be inconsistent on their own, it is expected that the whole dataset as it stands is inconsistent as well. The consistency test for the full dataset gives a consistency measure of -2.96, indicating an inconsistent dataset. What is of additional interest in the consistency analysis is the examination of the sensitivities of the different uncertainties in both the experimental measurements and model input parameters. Examination of the different Lagrangian multipliers can reveal if a particular data point is responsible for the inconsistency. For the current dataset there are three model parameters and 22 experimental measurements. Each of these 25 quantities has an upper and lower uncertainty bound. Each of these bounds has an associated Lagrangian multiplier. This gives a total of 50 multipliers for the sensitivity analysis. Figure 5.11 is a bar graph showing the values of the Lagrangian multipliers for the dataset. In this bar graph the first three multipliers correspond to the lower bounds of uncertainty for the thee model parameters: the helium flow velocity, the air co-flow velocity, and the system temperature. Bars 4 through 6 are the multipliers for the upper bounds of the same three model parameters. Bars 7 through 28 represent the values of the multipliers for the lower bounds of the uncertainties in all 22 experimental measurements, and bars 29 through 50 represent the multipliers for the upper bounds in uncertainty for the same measurements. The bar graph depicted in Figure 5.11 shows that one multiplier stands out above all the others. The 40^{th} multiplier corresponds to the sensitivity of the consistency measure to the upper bound of the uncertainty in dataset unit number 12. This corresponds to the first data point on the left-hand side of the plot in Figure 5.6. It is interesting to note that this point is one of the four points mentioned earlier that are inconsistent by themselves.

The next step in the analysis is to see how the consistency of the dataset is affected if the dataset unit associated with the most sensitive Lagrangian multiplier is removed. With the 12^{th} dataset unit removed, the consistency test is repeated for the reduced system. The result is a consistency measure of -2.47. The reduced dataset is still inconsistent. The sensitivity analysis of the reduced set is performed in the same manner as for the original set, and the resulting multipliers are shown in Figure 5.12. Again, one multiplier shows

a value much higher that any other. In this case it is the multiplier associated with the upper bound of uncertainty of the 13^{th} data point. This is the second data point from the left shown in Figure 5.6. Again, the system is showing the most sensitivity to one of the original four inconsistent points.

The above procedure is repeated. The 13^{th} dataset unit is removed from the dataset. The consistency analysis and sensitivity analysis are repeated. The resulting consistency measure is found to be -1.51. The newly reduced dataset is still inconsistent. The Lagrangian multipliers are shown in Figure 5.13. This time another multiplier value is shown to be far larger in value than the others. This multiplier is associated with the upper bound in uncertainty for the first data point, which is the first point plotted on the left-hand side in Figure 5.5. Yet again this point is one of the four inconsistent dataset units previously discussed.

The procedure of point elimination is repeated two more times. The dataset reductions lead to consistency measures of -0.72 with the first dataset unit removed, and 0.53 with the 22^{nd} dataset unit removed. With four of the original dataset units removed, the same four that were found to be inconsistent units, the reduced dataset is finally found to meet the criteria for consistency. The Lagrangian multipliers for the dataset with three units removed is shown in Figure 5.14, and the same plot for the set with all four inconsistent unit removed is shown in Figure 5.15. In each case one multiplier is found to have a much larger value than the others, and each is found to correspond to one of the four dataset units that was found to be inconsistent.

The point of this series of tests is not to demonstrate that any set can be found to be consistent if a sufficient number of units are omitted. The purpose is to determine if a set is consistent or not, and if it is found to be inconsistent to determine the source of the inconsistencies. The preliminary examination of the data plotted in Figures 5.5 and 5.6, the examination of the pair-wise consistency data in Figures 5.7 and 5.8, and the consistency and sensitivity analyses of the full system all demonstrate that the helium dataset is inconsistent because of the presence of four problematic dataset units. It is also shown that the consistency of the dataset is most sensitive to these four experimental points, specifically the upper bounds in the uncertainties of each of these points. The

higher value of the Lagrangian multipliers shown in Figures 5.11, 5.12, 5.13, and 5.14 do not necessarily justify the omission of these four dataset units, but they do indicate the need for the measurement of these points and the determination of their uncertainties be reviewed and reconsidered.

It is also interesting to note that at each stage of the consistency analysis there is some indication of the sensitivity of the consistency to the model parameters but never a dominating one. Though not illustrated further in this report, the analysis is continued by omitting further points whose sensitivity were the highest. In every observed case it is the upper bounds of the experimental uncertainty that display the greatest sensitivity and not the model parameters themselves. Also, there is always one multiplier that stands out above the others at each stage of the analysis. This is in contrast to a system where several multipliers show comparable value without a clearly dominant member. As already stated such a system would justify a second examination of the model itself, and the results seen thus far for the helium system do not indicate such a need.

5.6 Consistency Analysis: Final Iteration

Given the results from the first iteration of the consistency analysis the error bars on the inner and outer-most fringes are extended after review of the pinch and puff point refractive indices in these radial regions. At the pinch point the boundary layer of the plume contracts which results in pure air residing in a region that was previously mixed. During the puff cycle itself the puff volume of helium that emerges results in many gradients and thus many fringes emanating from its center. Figures 5.16 and 5.17 show the data points with the new error estimates. With the error bars for the inconsistent data points revised there is now overlap between experimental error and computational range for the four dataset units. A consistency analysis for the newly revised dataset shows that consistency is achieved.

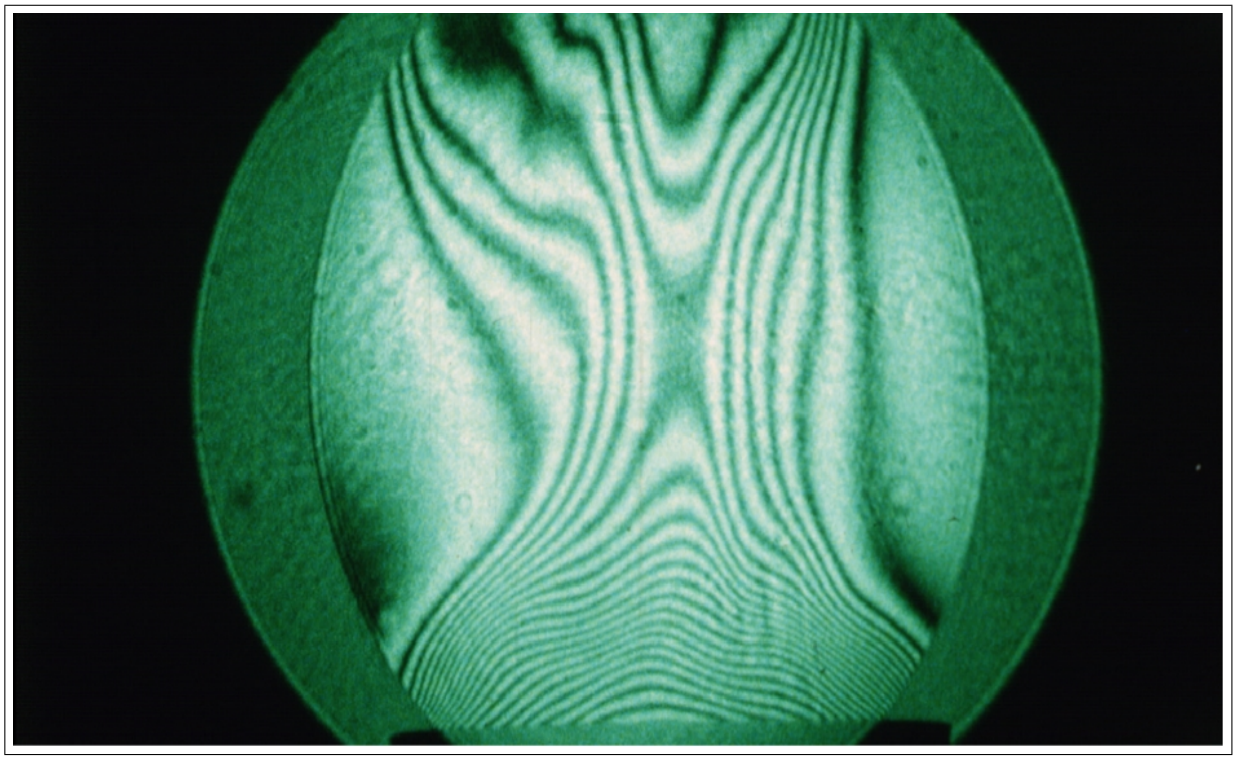

Figure 5.1: An example of an original interferogram of a helium plume.

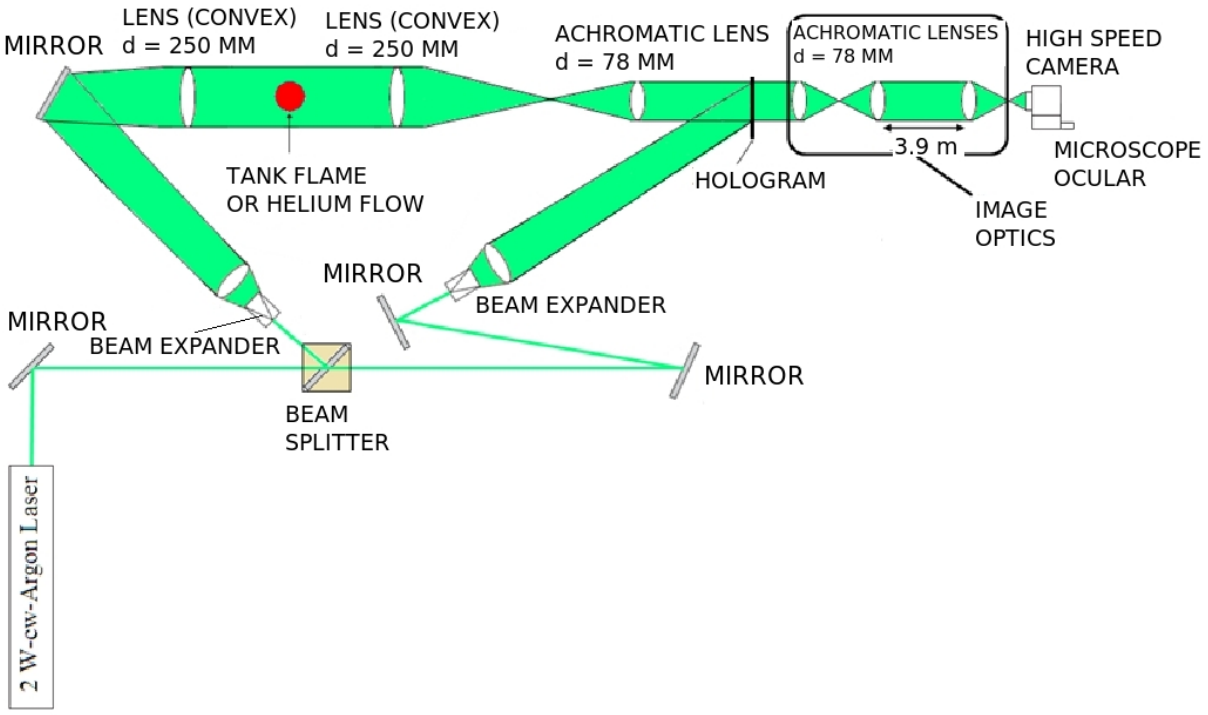

Figure 5.2: How holographic images of helium plumes are produced.

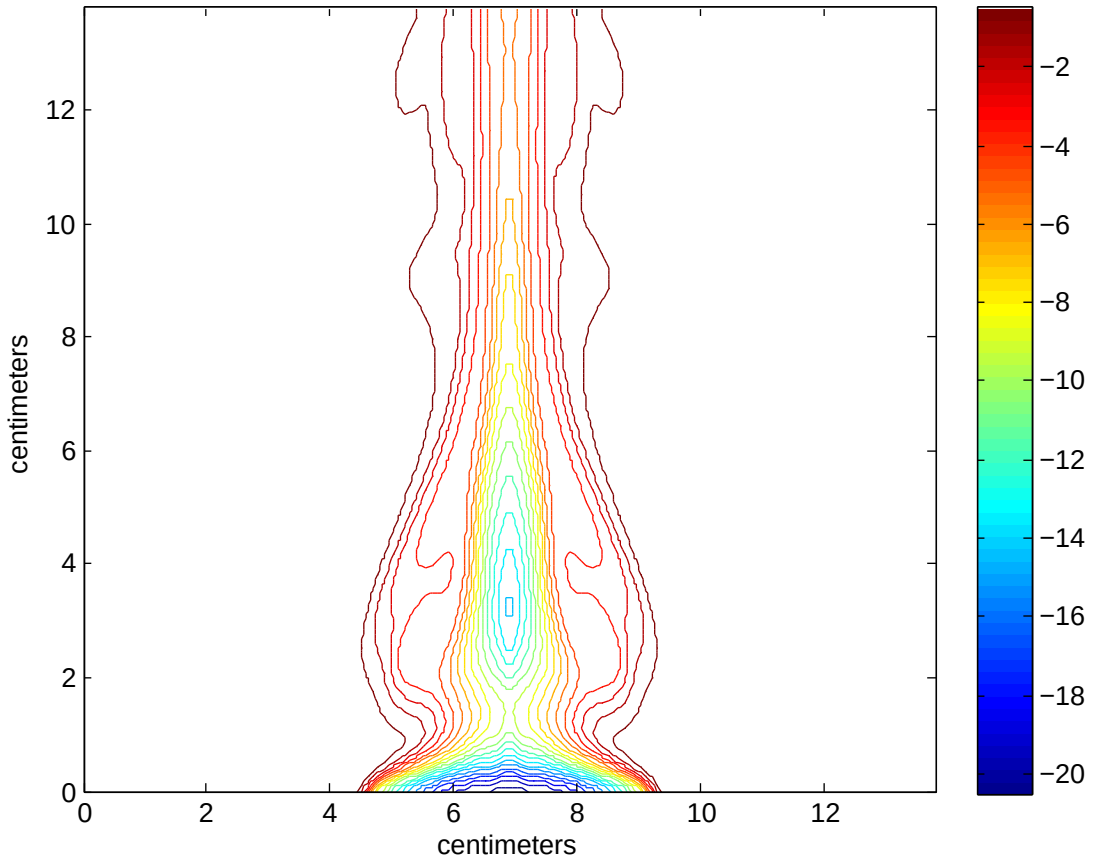

Figure 5.3: Interference order data from an ARCHES simulation of a helium plume.

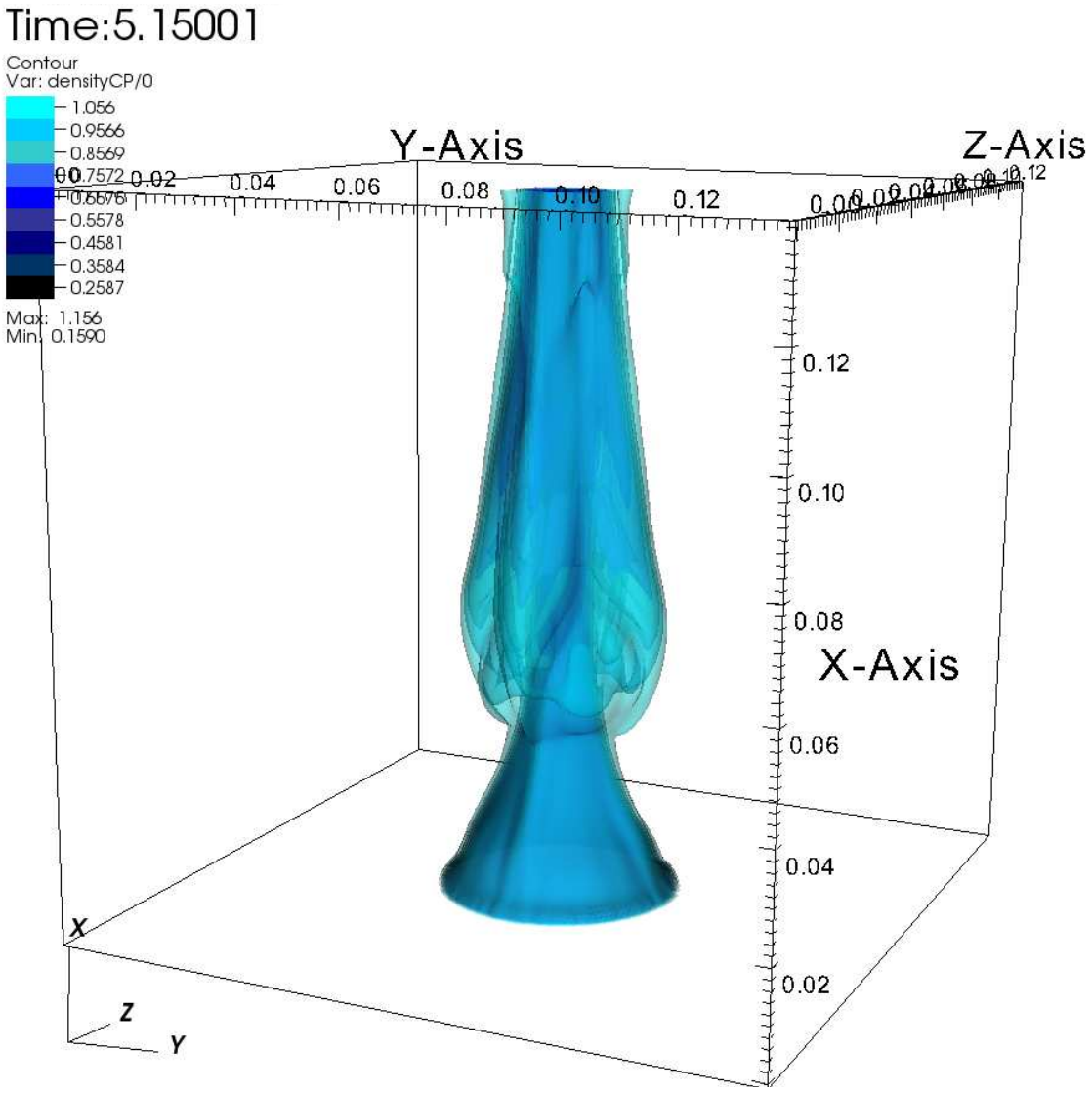

Figure 5.4: A density contour plot from a helium simulation in ARCHES.

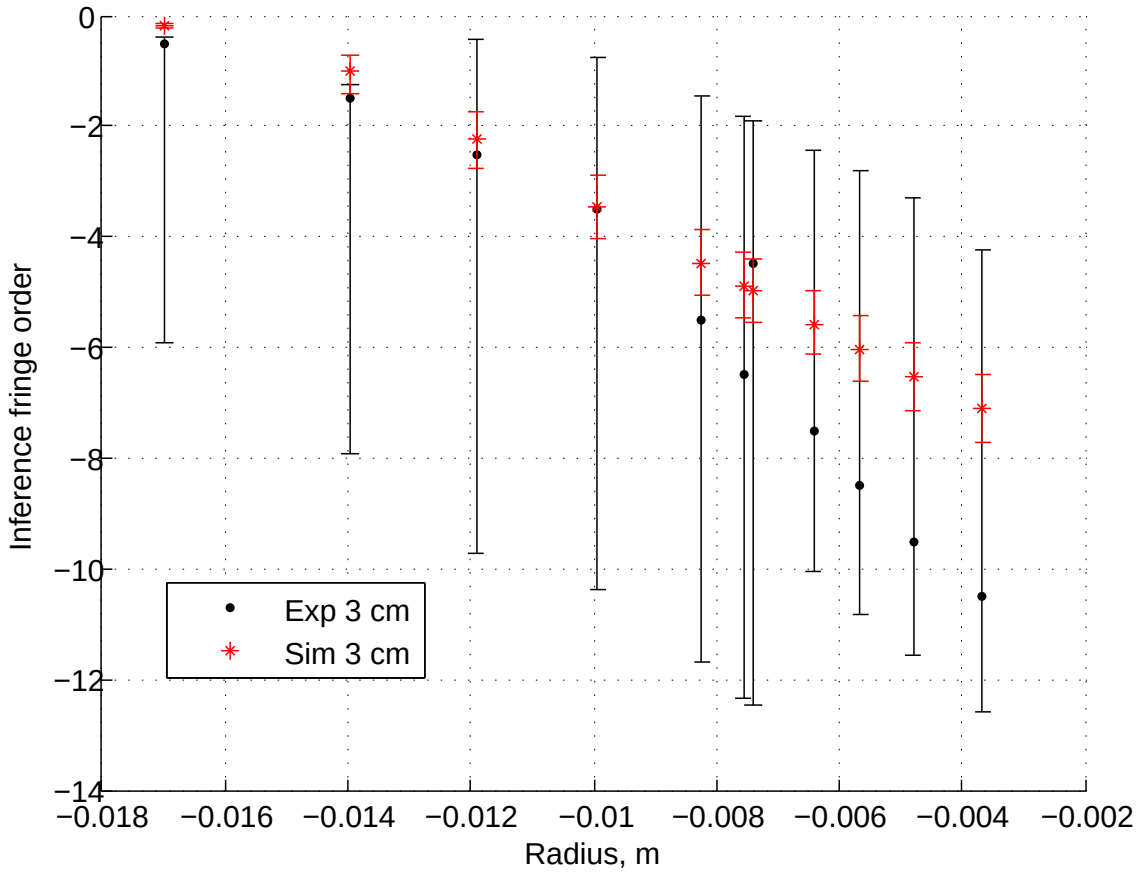

Figure 5.5: The data comparison of the interference order between experiment (in black) and computation (in red) at different radial positions at a height of 3 centimeters above the helium inlet.

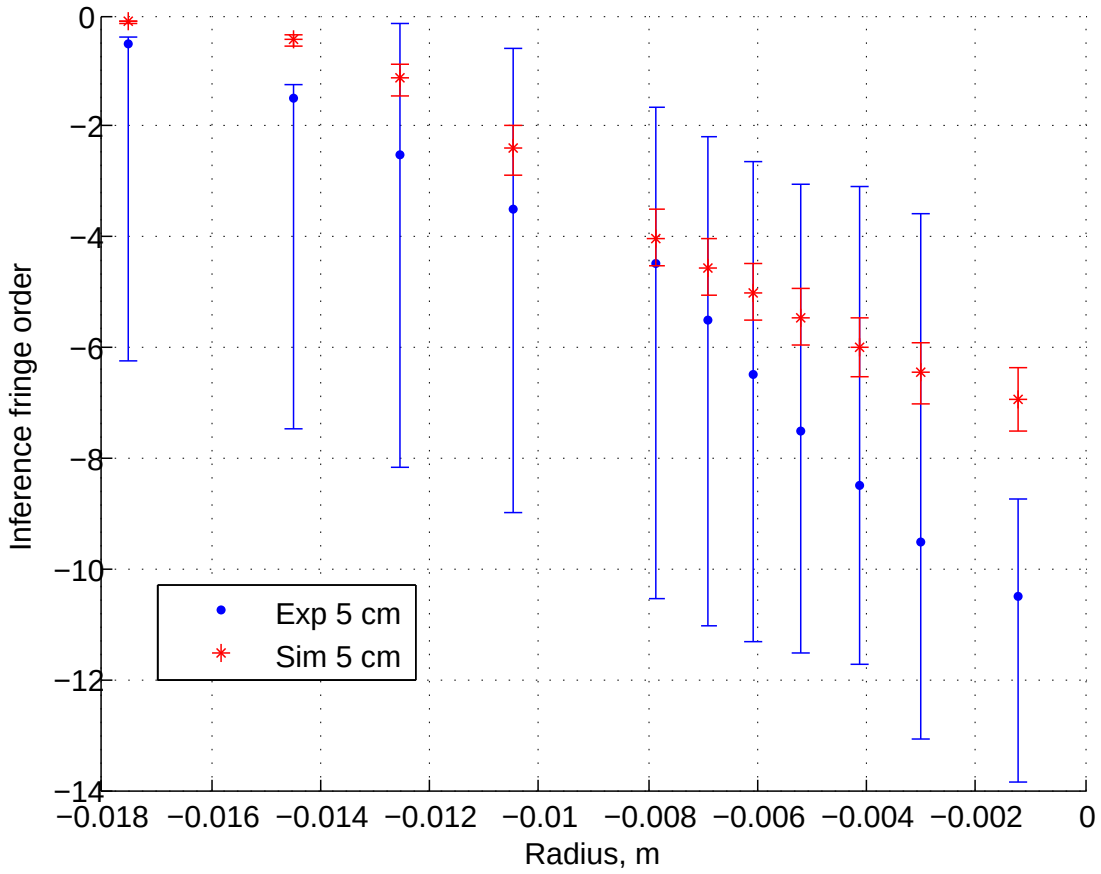

Figure 5.6: The data comparison of the interference order between experiment (in blue) and computation (in red) at different radial positions at a height of 5 centimeters above the helium inlet.

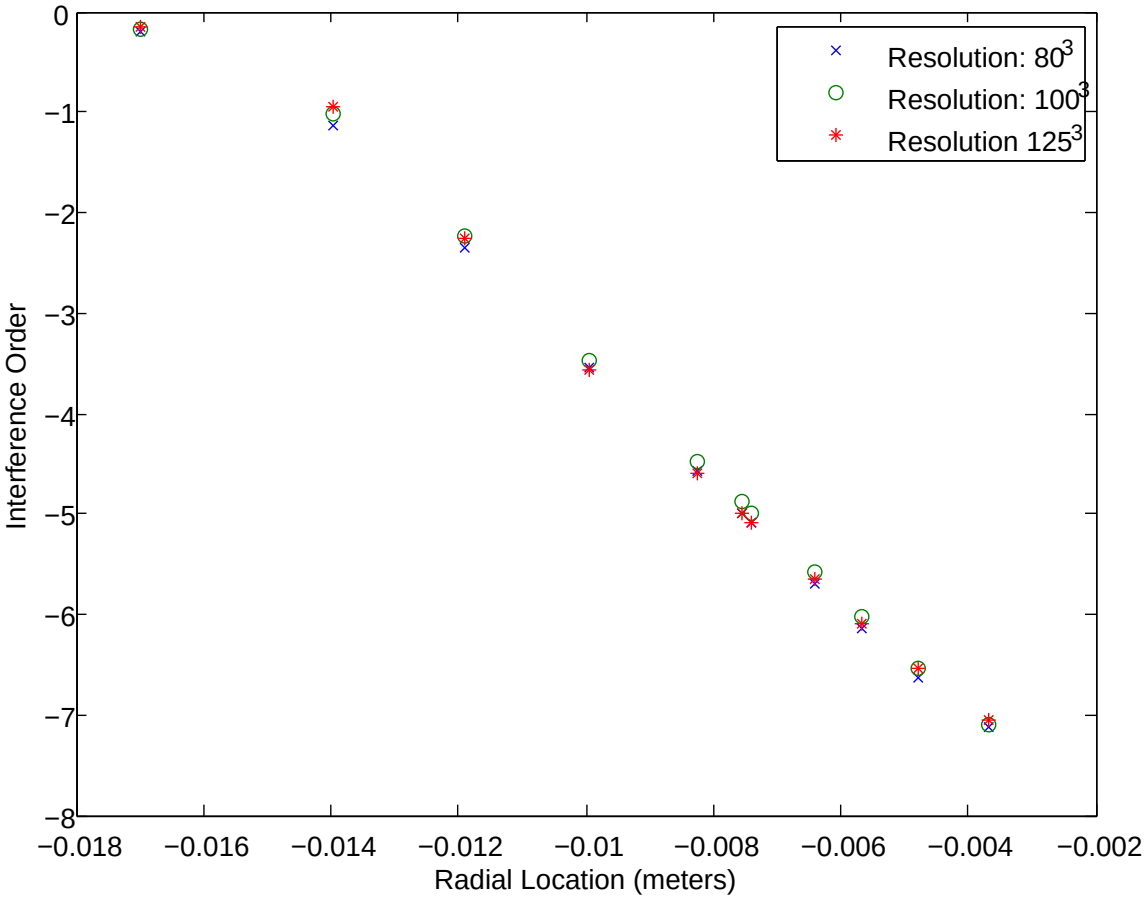

Figure 5.7: A comparison of simulation output from the base-case simulation run at three different resolutions for the data at 3 centimeters above the helium inlet.

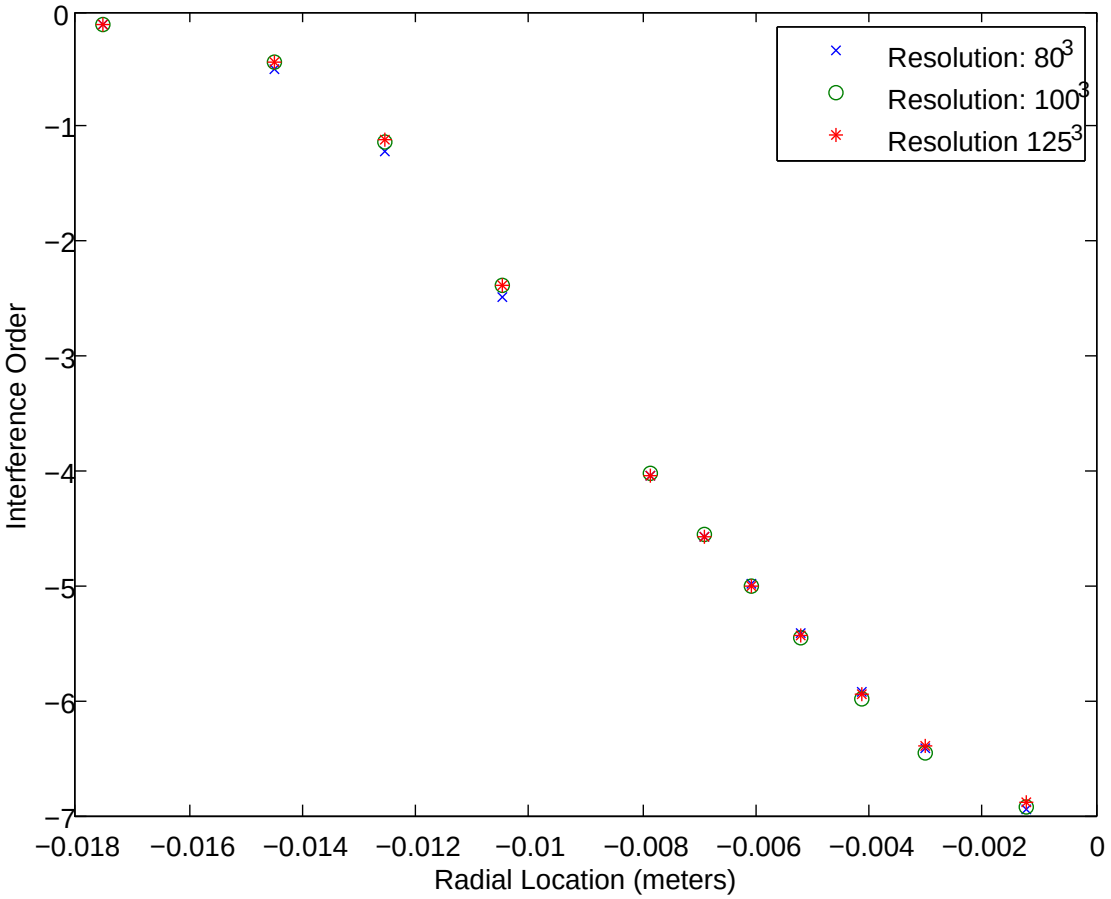

Figure 5.8: A comparison of simulation output from the base-case simulation run at three different resolutions for the data at 5 centimeters above the helium inlet.

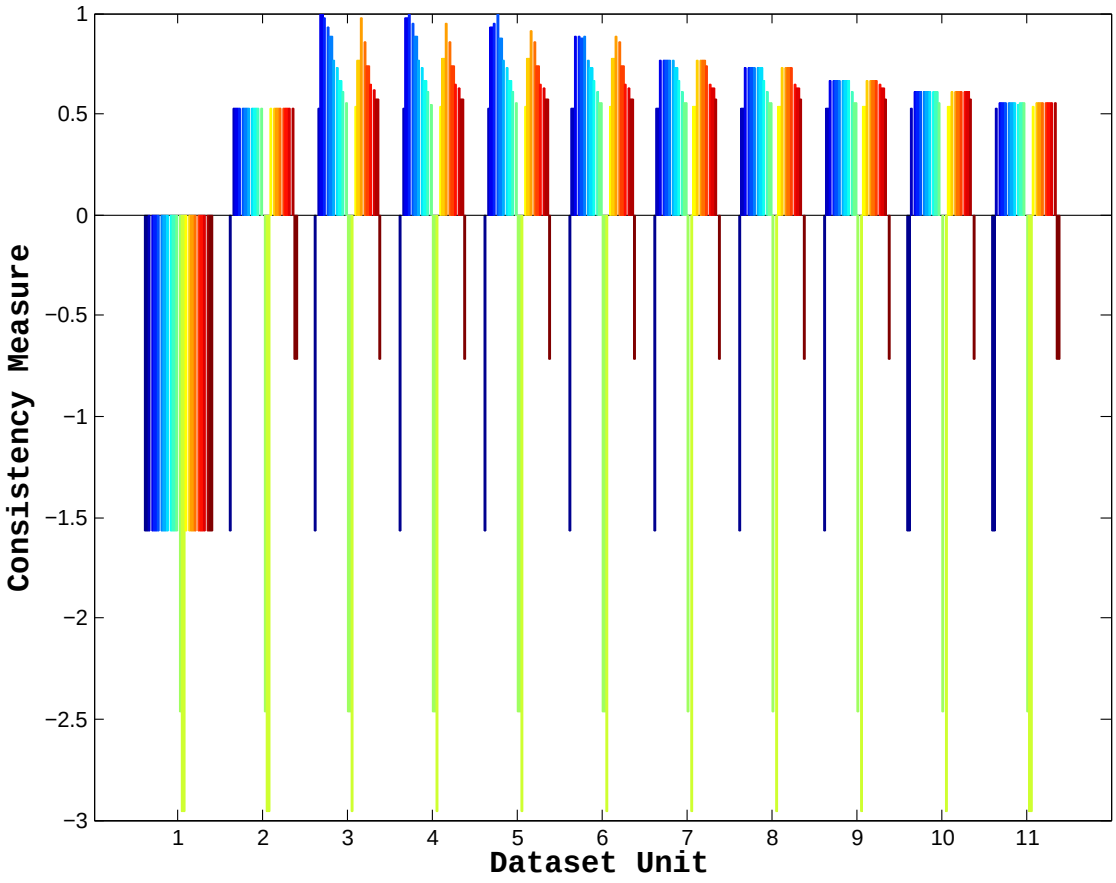

Figure 5.9: Consistency measures for all pair-wise analyses involving dataset units 1 through 11.

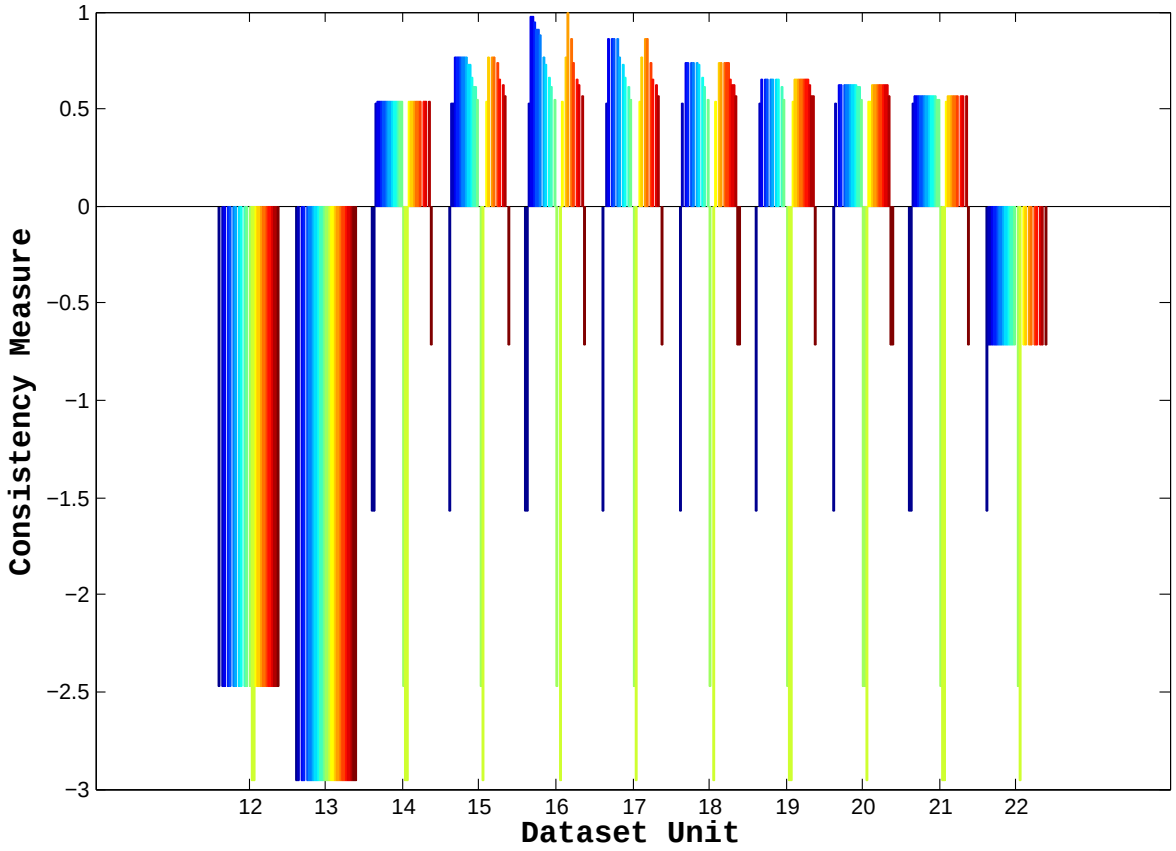

Figure 5.10: Consistency measures for all pair-wise analyses involving dataset units 12 through 22.

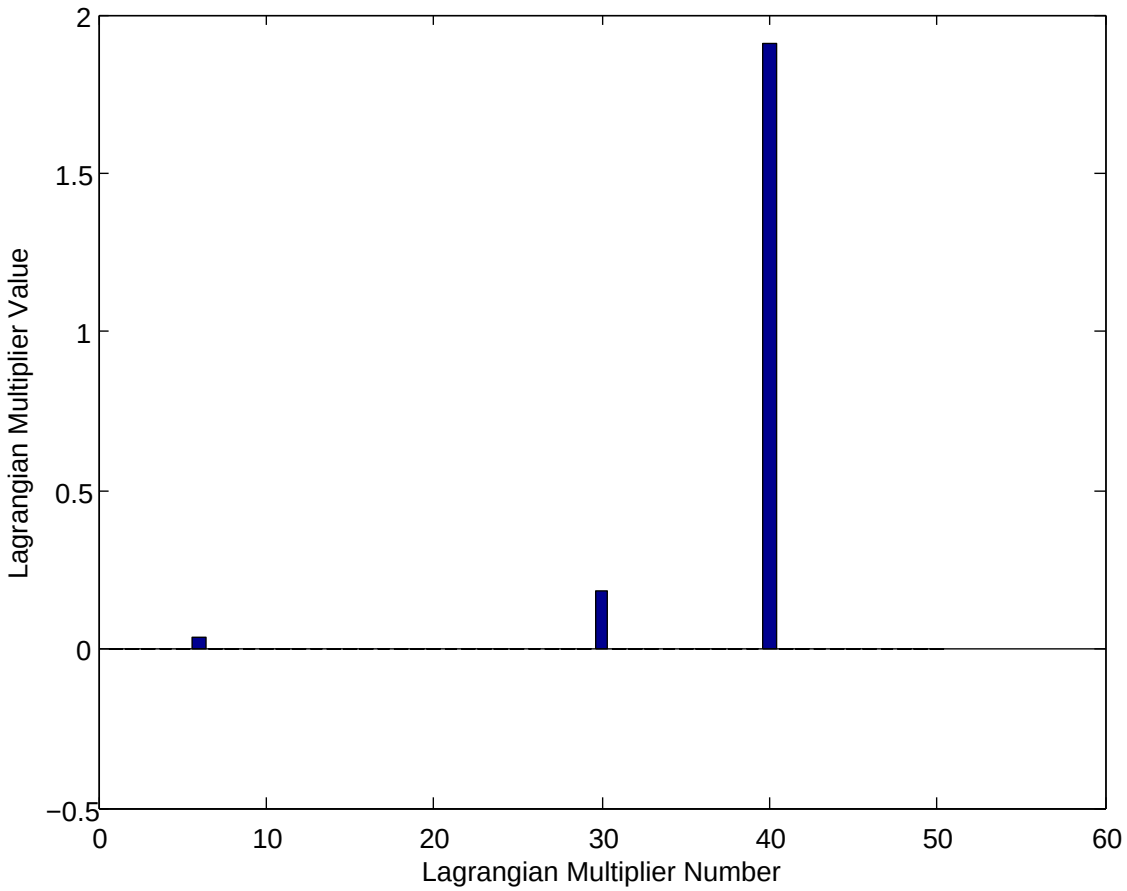

Figure 5.11: The values of the Lagrangian multipliers for the original helium dataset. The consistency measure for this set is -2.96, indicating an inconsistent set.

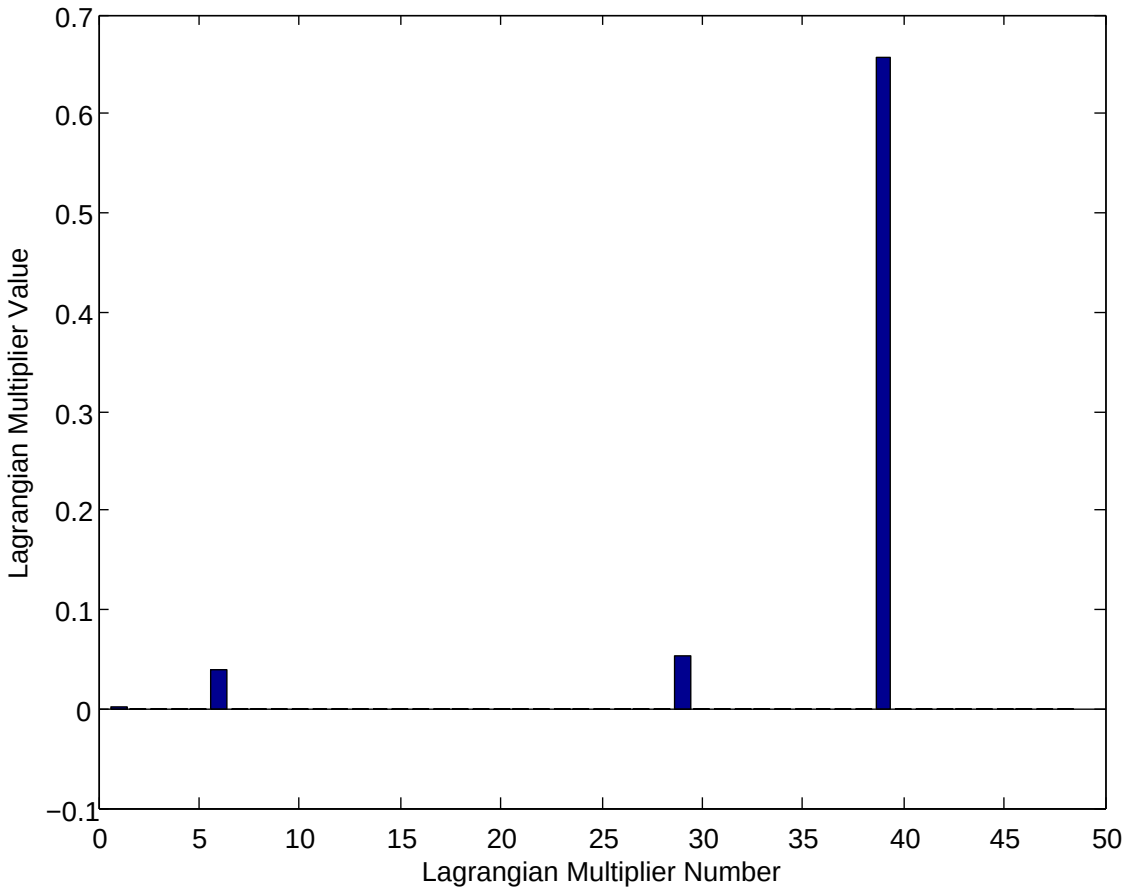

Figure 5.12: The values of the Lagrangian multipliers for the reduced dataset where one point (12) has been removed. The consistency measure for this set is -2.47, indicating an inconsistent set.

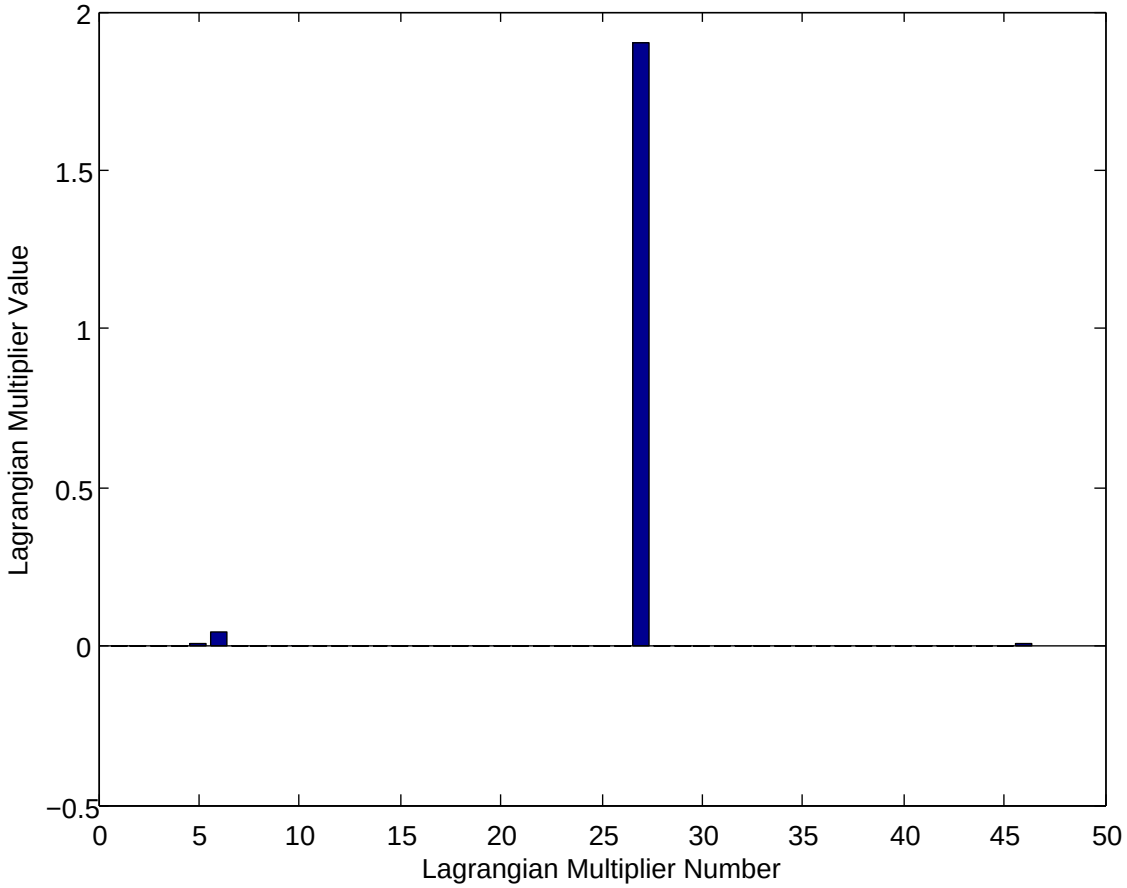

Figure 5.13: The values of the Lagrangian multipliers for the reduced dataset where two points (12 and 13) have been removed. The consistency measure for this set is -1.51, indicating an inconsistent set.

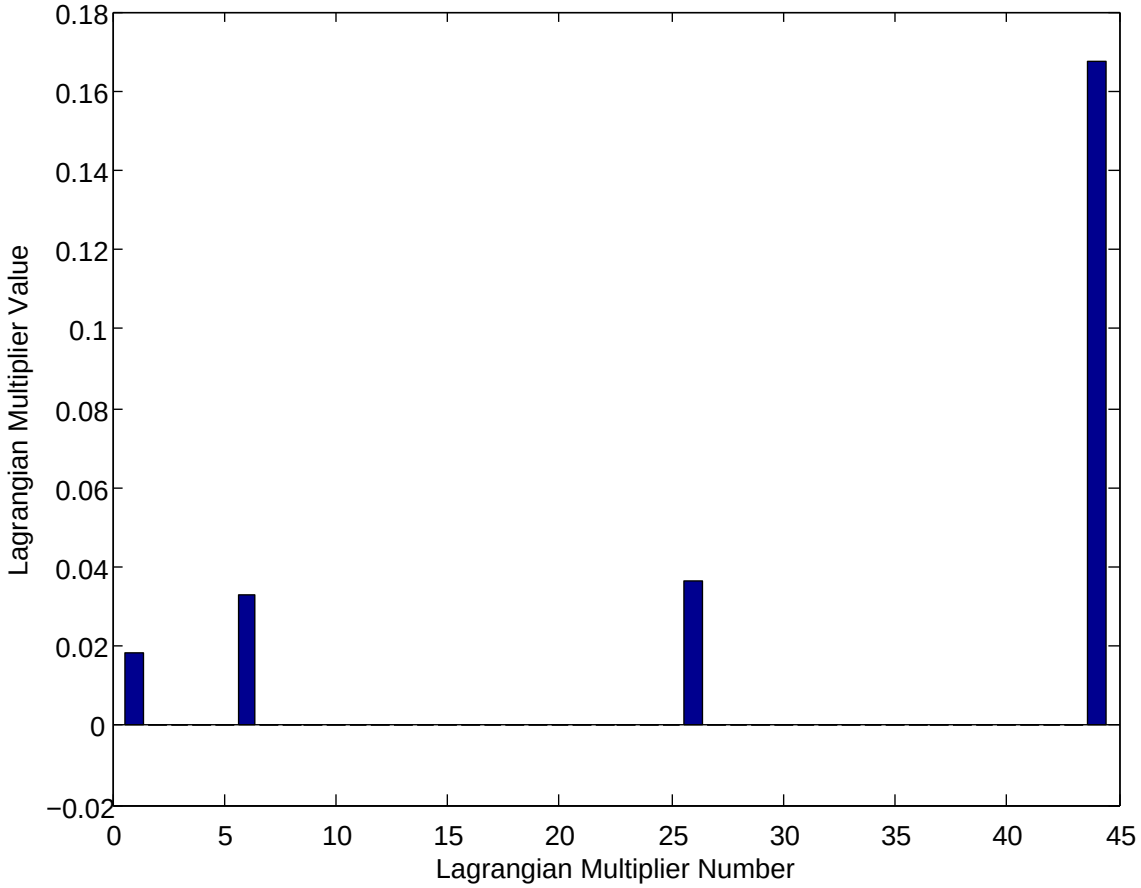

Figure 5.14: This bar graph shows the values of the Lagrangian multipliers for the reduced dataset where three points (12, 13, and 1) have been removed. The consistency measure for this set is -0.72, indicating an inconsistent set.

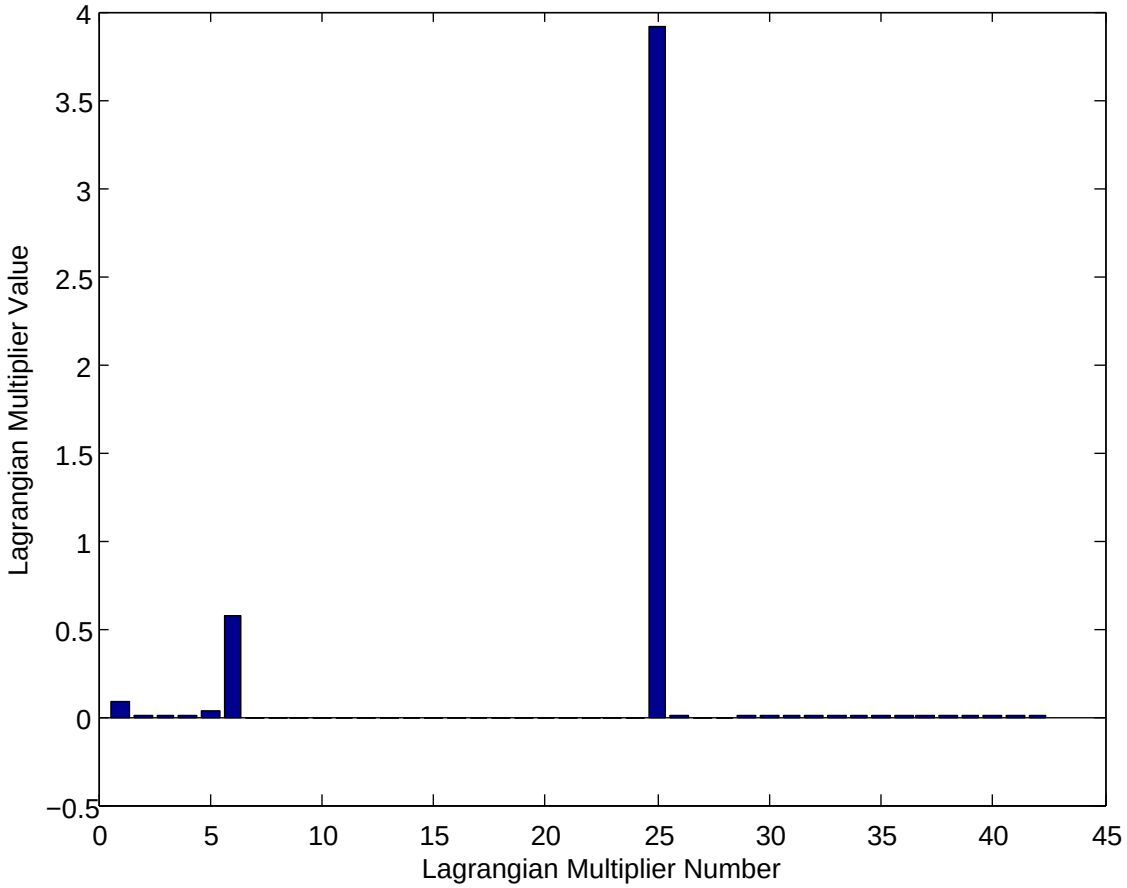

Figure 5.15: The values of the Lagrangian multipliers for the reduced dataset where four points (12, 13, 1, and 22) have been removed. The consistency measure for this set is 0.51, indicating a consistent set.

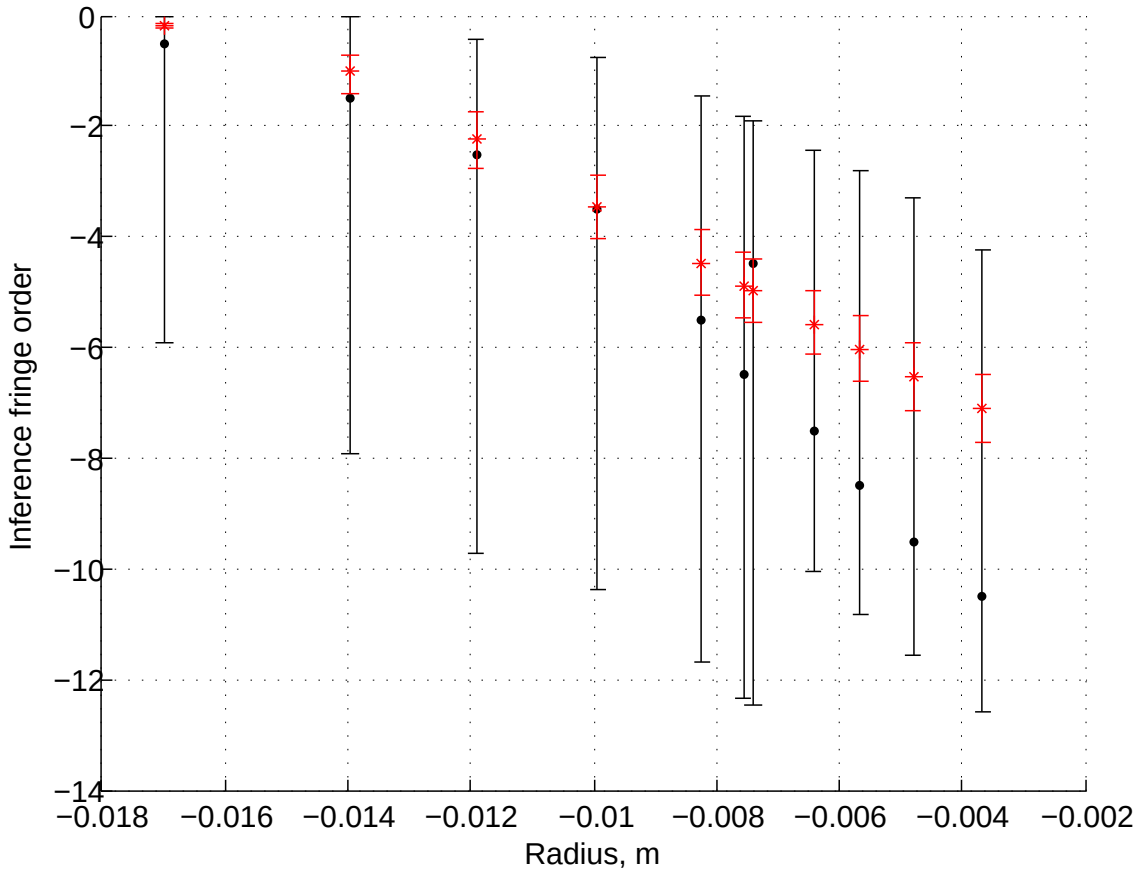

Figure 5.16: The same data as that seen in Figure 5.5 with the error bars revised.

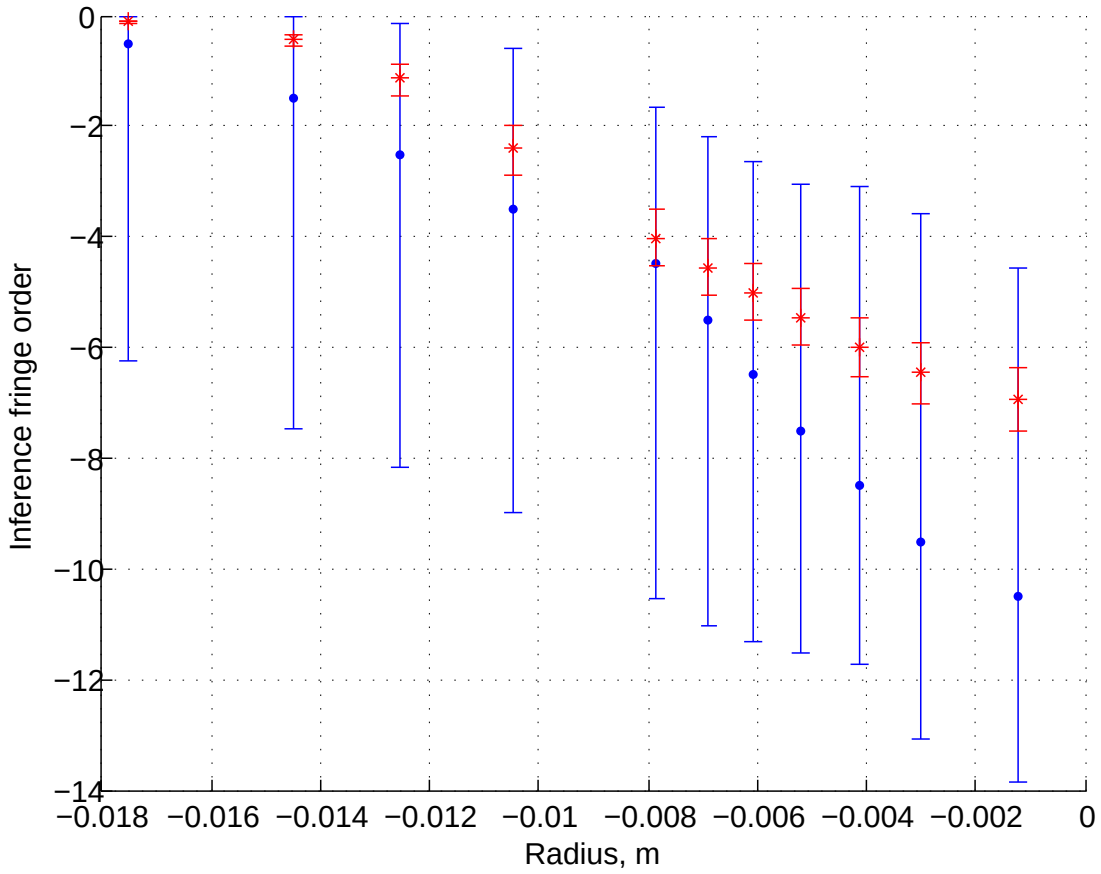

Figure 5.17: The same data as that seen in Figure 5.6 with the error bars revised.

Table 5.1: The proposed active variables and their relevant information for the helium plume simulations.

Input Variable	Uncertainty Range	Lower Bound	Nominal Value	Upper Bound
Helium Inlet Velocity (m/sec)	0.1215 - 0.1485	0.1215	0.1350	0.1485
Air Co-Flow Velocity (m/sec)	0.0135 - 0.0675	0.0135	0.0405	0.0675
System Temperature (K)	295.15 - 315.15	295.15	305.15	315.15

Table 5.2: The values for the inputs used in each simulation for the Box-Behnken designed helium plume simulations.

	System Temperature (K)	Air Co-Flow (m/sec)	Helium Inlet Flow(m/sec)
Case-Base	305.15	0.0405	0.1350
Case-001	305.15	0.0135	0.1215
Case-002	305.15	0.0675	0.1215
Case-003	305.15	0.0135	0.1485
Case-004	305.15	0.0675	0.1485
Case-005	295.15	0.0405	0.1215
Case-006	315.15	0.0405	0.1215
Case-007	295.15	0.0405	0.1485
Case-008	315.15	0.0405	0.1485
Case-009	295.15	0.0135	0.1350
Case-010	315.15	0.0135	0.1350
Case-011	295.15	0.0675	0.1350
Case-012	315.15	0.0675	0.1350

CHAPTER 6

ARCHES WITH THE RESPONSIVE BOUNDARY

6.1 Introduction

Chapter 4 describes the tests and procedures used to verify certain results that come out of the Responsive Boundary model. These activities quantifiy certain types of errors common to computational models. These include programmer error, iterative convergence error, and discretization error. Validation tests are applied to the Responsive Boundary model under limited conditions to demonstrate that the code is relatively robust and can generate output that is reasonable given reasonable model inputs. Full validation cannot be applied to the model on its own because certain inputs cannot yet be provided, nor can the model be fully tested under its intended use until it is linked to the ARCHES fire code. The purpose of validation is to evaluate and perhaps quantify error in the code that arises from the model formulation used to predict the physical system of interest. Such models, including those that are a part of the Responsive Boundary model, often incorporate approximations and estimations in order to make the model's solution obtainable. Chapter 3 describes the model used to predict pool fire behavior, and it describes several of the model's approximations and estimations. To evaluate the error from these sources, techniques like sensitivity analysis and data consistency are applied. Chapter 5 shows how these techniques can be used to evaluate model error in ARCHES simulations of helium plume data and how a collaboration study between ARCHES simulations and experimental data can give insight into the accuracy of both. In this chapter the focus returns to the performance of the Responsive Boundary model as it works with ARCHES. Both verification and validation methods are applied to a pool fire system model in ARCHES using the Responsive Boundary. This chapter completes the analysis illustrated

in the validation hierarchy in Figure 1.2 by discussing verification and validation activities concerning the top box in that hierarchy.

The first part of this chapter discusses some general trends and issues in the data generated from ARCHES pool fire simulations involving the Responsive Boundary. Next, grid convergence tests are applied to quantify error arising from spatial discretization in the gas-phase domain of the pool simulation and to determine the appropriate spatial resolution for pool fire simulations in the data collaboration phase of model evaluation. Lastly, a data collaboration study applies the results of Klassen and Gore (1992) for 30-centimeter pool fires as the experimental side of the collaboration and output from ARCHES and the Responsive Boundary model for the computational side.

6.2 General Observations

The data provided by Klassen and Gore (1992) have already been mentioned in previous chapters. Their data concerning 30-centimeter pool fires are the choice for simulation comparison in all remaining activities described here. Their data are used for verification and validation activities because they feature important measurements such as global mass burn flux for a number of fuels along with measurements of incident radiative heat flux to the pool surface. Blanchat et al. (2006) also give these types of measurements and they are more recent, but preference is given to the data from Klassen and Gore for a couple of reasons: (1) the pool fires of interest in their study are much smaller and therefore simpler to simulate compared to the 26-foot diameter fires that Blanchat et al. conduct, and (2) the fuels that Klassen and Gore use in their experiments are single-component and therefore less error is inflicted in approximating their fuel properties as opposed to the error that could arise from applying some type of single-component surrogate for the JP-8 fuel used in the study that Blanchat et al. conduct.

The computational system in ARCHES used for both verification and validation tests is a cubic volume with dimensions of 90 x 90 x 90 centimeters. The choice of 90 centimeters for each spatial dimension is in accordance with a rule of thumb in ARCHES simulations. The rule recommends a dimensional length of at least three times the inlet diameter in order to prevent simulation stability issues that arise when the buoyant plume comes too close

to the side boundaries of the domain. The fuel inlet boundary lies at the bottom-center of the domain, and it is at this boundary where the Responsive Boundary model is applied. The rest of the bottom face of the domain is a wall boundary condition. The four sides of the domain are assigned as pressure boundaries where the system pressure is defined just outside the computational domain. The top wall of the computational domain is an outlet boundary condition where the bulk of material arising from the buoyant pool fire is allowed to exit the domain.

The computational domain is divided into control volumes called nodes. The number of nodes depends on the desired resolution of the particular simulation. In accordance with the behavior of discretization error, the more nodes used in the domain the more accurate the predicted properties are from the simulations. In all simulations described here the number of nodes along each spatial dimension are kept equal. For example, the coarsest cases use 64 nodes in each direction. This means that for such cases the total number of computational nodes is $64^3 = 262,144$. The finer cases use even more nodes. The more nodes in a domain, the more nodes there are adjacent to the inlet boundary. This means that the pool surface is divided into more surface areas. For each computational surface region at the inlet there is a different value for the incident radiative heat flux, the gas temperature, and the gas composition, and for each surface region there is an instance of the Responsive Boundary model and a prediction for the mass flux. A greater number of nodes used in a particular computational domain implies more accurate data are supplied as input to the Responsive Boundary model, and therefore more accurate mass flux predictions arise from the boundary model.

6.3 Simulated Pool Properties

With the boundary model linked to ARCHES initial simulations are run to test the robustness of the computational set up and to see if the results are consistent with experimental observations of pool fires. This section reports some of the general findings in regard to the pool properties that most affect the boundary model and its performance. The figures shown in this section arise from simulations of the 30-centimeter heptane pool fire. The resolution used in these simulations is 200^3 nodes. The code framework

developed for ARCHES employs certain methods and data types for storing and organizing computational results. The data stored are usually physical properties such as enthalpy, temperature, composition, and thermal radiation fluxes. Because ARCHES models three-dimensional domains, these physical properties are stored as three-dimensional type arrays. In contrast, the properties of interest with the boundary model are not defined on a three-dimensional domain, but they are defined at the pool surface, a two-dimensional domain, and some adjustment to the code is necessary to accommodate the tracking of these physical properties. These properties include incident radiative heat flux to the pool surface, convective heat flux the to pool surface, pool surface liquid temperature, and mass flux of fuel from the pool surface.

The fuel mass flux from the pool surface is the property of primary interest as it is the property associated with the purpose of the development of the Responsive Boundary model. For the validation tests with the stand-alone boundary model a full measure of this property is not possible because certain inputs are not available at that stage. With the boundary model linked to ARCHES a more complete realization of this property is possible. Properties like spatially varying radiative heat flux and the presence of fuel in the gas phase just above the pool surface allow a more realistic prediction of the behavior of the pool with regard to its evaporative mass flux. Figures 6.1 and 6.2 are color plots of the mass flux at the pool surface. Each plot of data is sampled from a different time in the evolution of the pool fire. Figure 6.1 shows the mass flux data at one point in the transient heating stage (at time = 0.5 seconds) of the pool fire. Figure 6.2 show the same data at a point where steady burning is considered to have been reached (at time = 7.0 seconds.) Figure 6.1 shows that during the warm-up phase of the fire that the magnitudes of the mass flux are relatively low (less than $0.001 kg/m^2 - sec$) and that the regions of highest flux are at the edges of the pool. The fluxes decrease moving from the edge of the pool toward the center. There is a region around the center of the pool where no evaporation is happening. This is due to the presence of high fuel composition in the gas phase just over the pool surface (about 2.25 millimeters over the surface.) The fuel composition data are plotted in Figure 6.3 for the transient time period. Figure 6.2 shows that for the steady burning period a very different pattern arises in the mass flux data. In this case the

entire pool is contributing to the overall flux. The figure shows that there are regions of higher flux (indicated in red) that cover large sections of the pool near the center. These regions of higher flux migrate around the center of the pool as time progresses, but the general pattern is that flux is highest in the central regions of the pool surface with a slight decrease in flux at the pool's edges. This is the opposite pattern seen for the flux data at the transient phase. The regions of higher mass flux in Figure 6.2 correspond to regions on the pool surface that recieve more thermal input at that point in time. Since radiation heat flux does not distribute evenly over the pool surface, the mass flux also has non-uniform regions of different flux. Figure 6.4 shows the gas phase fuel composition for the steady phase. Figures 6.3 and 6.4 show that there is a significant presence of fuel over the surface of the pool during the transient phase and a dominant presence of fuel for the steady phase. The fuel's presence is strongest near the center of the pool, and this explains why there is no evaporation at the center of the pool for the transient period. At this time of the fire the pool's liquid surface is relatively cool as shown from the surface temperature data in Figure 6.5. At this point the fuel's vapor pressure is not sufficient to overcome the fuel barrier at that location. In contrast, despite an even stronger presence of fuel in the gas phase at the steady period, the entire pool evaporates fuel at a higher rate. The flux rate is a function of liquid temperature, and at the steady burn phase the liquid warms close to its boiling point (for heptane the boiling point is 371.59 Kelvins) as shown in Figure 6.6. At this temperature the fuel's vapor pressure is high enough to overcome the fuel barrier and volatilize over the whole pool surface.

In addition to the the mass flux, the different heat fluxes to the pool are also of interest as they directly affect the mass flux rate. Figures 6.7 and 6.8 show the radiative heat flux to the pool surface at the transient phase and the steady phase, respectively. Figure 6.7 shows for the transient burning phase the radiative flux is strongest at the center of the pool and decreases in magnitude toward the edge. Figure 6.8 shows a similar pattern for the radiative flux at the steady burn phase as the pattern shown in Figure 6.2 for the mass flux. At the steady period the regions of highest radiative fluxes cover the central areas of the pool, and the flux decreases somewhat at the edge of the pool. This observation is consistent with that of Klassen and Gore (1992) for the radiative heat flux patterns over

the pool surface for 30-centimeter heptane pool fires. The similarity of pattern between mass flux and incident radiative heat flux suggest a correlation between the two at steady burning. This is reasonable since the heat input to the pool is what determines the mass flux rate, and radiation is thought to be the dominant mode of heat transfer at steady burning for pools of this size. This similarity of pattern is not found for the transient phase as is verified through comparison of Figures 6.1 and 6.7.

The other heat flux of interest is the convective heat flux. Figures 6.9 and 6.10 show the data for this quantity at transient and steady burning phases, respectively. Both plots show that the spatial distribution pattern for convective heat flux is opposite that for radiative heat flux. Convective heat flux is strongest near the edges of the pool and decreases toward the center. As has already been described, convective patterns follow the location of the pool fire's flame sheets. As the sheets tend to emanate from the edges of the pool it is expected that the convective heat component is strongest there. Another important observation with the behavior of convective input to the pool is that the convection is strongest during the transient phase of the pool fire. At steady state the convective heat fux is negligible everywhere except at a thin region at the pool's edges. By the time of steady burning the fuel vapor flux has pushed the flame sheets so high that the temperature of the gas near the pool's surface is actually less than the surface liquid temperature throughout the central regions of the pool surface. Figures 6.11 and 6.12 show side views of the pool fire domain. The plotted data include the gas temperature, and the regions of highest temperature are assumed to be the locations of the flame sheets. The view in Figures 6.11 and 6.12 is a two-dimensional slice that goes through the center of the pool. Figure 6.11 shows that for the transient burning phase the flame sheets do indeed come close to the pool surface as expected, and Figure 6.12 shows the flames sheets largely separated from the pool surface except near the edges.

The primary output of interest in the pool fire simulations is the fuel mass flux at steady burning conditions. The data just examined show that although convective heat flux may play an important role in heating the pool during the transient stages of the fire, its influence is negligible at the steady phase. The radiative flux is what controls the mass flux at steady conditions, therefore, both mass burn rate and radiative heat flux are of

interest in subsequent analysis.

6.4 A Computational Issue

Before the verification and validation tests can be executed a computational issue needs to be addressed. As already stated the outputs of interest in subsequent analysis are the mass flux and heat input to the pool at steady state. The fact that these properties are desired at the steady burning phase presents a problem that challenges time constraints for the project. ARCHES simulations of pool fires using the spatial resolution described here require a substantial amount of computational resource and time to run. Typically one to two weeks of daily simulations are required to generate 10 to 12 seconds of simulation data for the finest resolutions used here. Depending on the size of the fire and the fuel burned, it takes 30 to 60 seconds to reach steady state. To further illustrate the problem Figure 6.13 shows a plot from a 30-centimeter heptane fire simulation. The simulation in question uses 150^3 nodes and is run for about twenty 12-hour periods using 224 processors. A total of 30 seconds is simulated. The plot in Figure 6.13 shows the global (over the whole pool surface) mass flux of fuel as a function of time. Examination of the plot shows that after 30 seconds the pool is barely coming to a steady behavior, and more time is needed to confirm the steady values for the global mass flux and other properties.

The simulation performance is an issue because of time constraints. For the purposes of grid convergence and data collaboration tests numerous simulations like the one just described are to be performed. Many of them are as big or larger than the one shown in Figure 6.13, and to be of any use they must all be run to steady state. Under these conditions there is not sufficient time to perform all the needed simulations.

The solution to this problem lies in the way the Responsive Boundary model performs its time stepping. In normal operation ARCHES calculates an appropriate time step to advance its predictions, and this time step is passed to the boundary model so it can keep pace with the fire code. Instead, the time step passed to the boundary model is multiplied by some reasonable amount and the boundary model advances faster. In this case the time step is multiplied by 10. In this situation the simulation proceeds normally, except that the pool responds 10 times faster to the input from the flames. The pool heats up 10 times

faster and steady state is hopefully reached much faster. The physics of the gas phase domain remains at normal speed while the behavior of the pool accelerates. Experience shows that the time to peak radiative flux is much faster than the time to steady mass flux. Figure 6.14 is a plot of the spatially averaged radiative heat flux to the pool surface for the same simulation that Figure 6.13 is based on. For this simulation, the radiative flux levels reach steady value with a second or two. It is the heating of the pool that holds up development of the mass flux, and accelerating the response of the pool can shorten the time to steady state and make the planned verification and validation tests tractable. Figure 6.15 show the same type of plot as that shown in Figure 6.13. In this simulation conditions are the same as before except that for the first 3 seconds of simulation time the pool's response is accelerated by a factor of 10. This allows 30 seconds of pool response to happen in the first 3 seconds of simulation time. After 3 seconds of simulation the code operates normally without acceleration. Figure 6.15 shows that the transient warming period that took almost 30 seconds in Figure 6.13 now requires only 3 seconds, and a steady behavior in the global mass flux occurs by about eight seconds of simulation time.

If phenomena related to the transient burning phase were of interest then the acceleration applied might not be acceptable, but since it is the steady properties that are of interest, the acceleration allows access to those properties in a much more workable time frame. Additionally, it is found in subsequent simulations that 3 seconds of pool acceleration are not sufficient to reach a quick steady behavior for certain simulations. Therefore, subsequent simulations are largely run with pool acceleration for the duration of the entire simulation. This is the reason that the data shown in Figures 6.1 to 6.12 features simulation results that are described as steady burning phase data from a simulation time of only seven seconds. The most prominent observed side effect of the acceleration is that the mass flux oscillates more rapidly as it responds to the oscillatory nature of the radiative heat flux, but the average value for the mass flux should remain unaffected.

6.5 Grid Convergence Tests

The grid convergence tests described in Chapter 4 test the spatial and temporal grid convergence error for the liquid pool model. With the model linked to the fire code the next

step is to perform the appropriate verification tests for the combined computational system. The application of a manufactured solution to test the reliability of the ARCHES code is not necessary for this present study as such tests have been previously applied. What is necessary is to test the effects of spatial discretization error in the gas phase domain on the intended output of the pool fire simulations. Time step discretization is not tested because the time steps utilized in ARCHES are quite small, and therefore, the error arising from temporal discretization is considered negligible. To test the spatial discretization error a series of simulations using different spatial resolution is run. Again, these simulations are based upon the heptane and methanol pool fire experiments of Klassen and Gore (1992). The dimensions and parameters of the computational domain were described earlier. The purpose of this series of tests is to gauge the discretization error and account for it before attempting to analyze modeling error and to determine an appropriate spatial resolution for the pool fire simulations that is fine enough to minimize the discretization error and not so fine that future simulations become too computationally expensive. The proper spatial resolution, once determined, is used in the validation test simulations.

For the heptane re, seven simulations are performed at different resolutions in order to find a convergent resolution. The coarsest resolution is 64^3 nodes and each subsequent simulation increases the dimension by 25% over the previous run (so the dimension of each cases is 64^3, 80^3, 100^3, 125^3, 150^3, 195^3, and 245^3.) Figure 6.16 shows a plot of the global mass burn rate with time for each of the seven heptane simulations. In each case a steady behavior is reached in the mass burn rate. Figure 6.17 shows a plot of the spatially averaged radiative heat flux to the pool surface with time, and Figure 6.18 shows a plot of the spatially averaged convective heat ux to the pool surface with time. Figures 6.16, 6.17, and 6.18 all arise from the same series of simulations.

Additionally, a similar series of grid convergence simulations are performed for the methanol pool fire. This series of tests uses similar resolutions as the heptane tests, except that there are no tests using a resolution of 245^3 or 64^3 nodes (only five resolutions for the methanol pool fire). The global mass burn fluxes for this series of tests are plotted in Figure 6.19, the radiative heat flux is plotted in Figure 6.20, and the convective heat fluxes are shown in Figure 6.21.

Despite the differences between heptane and methanol fires, the trends in the grid convergence results between the two types of pool fires show certain similarities. The first important observation is that the mass burn rate tends to follow the same pattern as the radiative heat flux for both fires during the steady burning period. This means that if the steady radiative heat flux finds convergence, then the steady mass burn rate does as well. This observation is consistent with previously stated dependencies of the burn rate on the radiative heat input in the literature. Another observation of importance from the data shown in Figures 6.16 through 6.21 is that both quantities find convergence somewhere between 195^3 and 245^3. Based on this finding, a resolution of 200^3 nodes is used for the next series of tests. In this test the expected order of convergence and the use of the GCI are problematic. Because the steady burn rate follows the radiative heat flux, its order of convergence follows that of the radiation data. The equations utilized in ARCHES that describe radiation heat transfer are of a different sort than the continuum equations that describe energy, mass, and momentum conservation in both the fire code and in the boundary model. The discrete ordinates methods used to solve the radiation transfer equations give rise to error whose order of convergence is not known in this application. For the three finest grid convergence simulations in the heptane fire analysis the GCI for the global mass flux data suggests an order of convergence of 4.40. Since the proper order of convergence for this situation is unknown, it is impossible to comment on the appropriateness of this figure. Since the measured order of convergence is higher than one, it is reasonable to conclude that the solution from the simulation approaches the true solution to the continuum models as the spatial discretization length approaches zero. Also, the relatively small changes in the steady global mass flux and the steady radiative flux between the two finest cases (resolutions of 195^3 and 245^3) as seen in Figures 6.16 and 6.17 give good reason to conclude that in this range of resolution the spatial discretization error is small compared to the modeling errors yet to be evaluated.

Another finding worthy of note comes from Figure 6.18. In this plot it is evident that while convection plays an important role during the early transient stages of the pool fire, its influence falls away and becomes negligible at the steady burning period. The same plot for the methanol fire simulations in Figure 6.21 shows a similar pattern. Although

somewhat higher for methanol the convective contribution is negligible compared to the radiative contribution, even for methanol. These findings are consistent with the pool data shown previously in Figure 6.10 and with the discussion on flame sheet behavior during the different phases of pool fires.

Another important trend in the data shown in Figures 6.16 through 6.21 is that the radiative heat flux is predicted higher at coarser resolutions for both heptane and methanol pool fire simulations. As the resolution becomes finer the predicted radiative heat flux diminishes. As steady mass burn rate depends directly on radiative input, its data show the same pattern. The mass burn rate data for both heptane and methanol are shown in Figures 6.16 and 6.19, respectively. In these plots the experimentally measured values for the steady mass burn rate as reported by Klassen and Gore (1992) are also included and are shown as straight, horizontal lines. The two values for each pool fire come from different experiments. The radiative feedback data that they measure and that are used for comparison arise from the experiment with the lower mass burn rate value. The higher mass burn rate measurement will be largely ignored from this point on.

Before the discussion of the data consistency analysis, another important feature of the data needs to be mentioned. The value of the spatially averaged radiative heat input at steady conditions for the heptane re simulation, based on Figure 6.17, is about 25 kW/m^2 (for the nest cases.) Experimental data for this same quantity as measured by Klassen and Gore suggest a value of about 18 kW/m^2. Also, the mass burn rate predicted from the highest resolution cases is predicted to be about 0.03264 $kg/m^2 - sec$. This value is slightly less than the lower measurement offered by Klassen and Gore, 0.0331 $kg/m^2 - sec$. A similar pattern is also seen with the methanol data shown in Figure 6.19 and 6.20. For the the methanol fires, simulations suggest an average steady radiative input of about 15 kW/m^2 compared to an experimental average of about 10 kW/m^2. The predicted mass burn rate of 0.0122 $kg/m^2 - sec$ is close to the experimental value of 0.0115 $kg/m^2 - sec$. In both cases there is good agreement between simulation and experiment with mass burn rate, but the radiation data are over-predicted in the simulations for both fuels. As mass burn rate and incident radiative heat flux to the pool are so tightly correlated this suggests a potential problem with consistency. This problem is discussed further in the next section.

6.6 Consistency Analysis

With the discretization error accounted for, the next step is to determine the effects of modeling error via a consistency analysis as described by Feeley et al. (2004). The first step in such an analysis is to determine which model parameters are most error prone and most likely to effect the model output. The model outputs include the steady global mass burn rate and the incident radiative flux to the pool surface. Because of time constraints only the heptane pool fire system is considered in this series of tests.

In the Responsive Boundary model a number of assumptions and physical property estimations are thought to contribute to the error in the mass burn rate predicted from the model. The physical properties of interest include fuel gas phase density, viscosity, thermal conductivity, and diffusivity. These properties are used for film characteristics between the liquid and gas phase, and they all contribute to the value of the mass transfer coefficient. This parameter is central to the predictions of the mass burn rates from the Responsive Boundary model, and the correlations used to obtain this parameter are also thought to be error prone. Another estimated property that is thought to have error is the liquid density. A simple test has shown that a $\pm$ 11% variation in liquid density produced a negligible variation in the mass burn rate. The extinction coefficient of the liuqid fuel employed in the boundary model determines the character of thermal energy absorption in the top liquid layers and therefore the temperature profile as well. It is thought that there is significant uncertainty in this parameter as well. A similar test to that done for the liquid density shows that significant changes to the extinction coefficient has a neglibible effect on the steady mass burn flux. For the purposes of consistency analysis an error of $\pm$ 15% is assumed for the mass transfer coefficients produced by the Responsive Boundary model. This estimate of the error is based on the error assumed for the heat transfer correlation developed by Dittus and Boelter (1930).

Another source of error thought to be of importance arises from a model used in ARCHES. As discussed already, the values of radiative heat input to the pool surface predicted by ARCHES are of concern. An important quantity that affects the radiation values predicted by ARCHES is the soot content (expressed as soot volume fraction). Since soot particles are largely responsible for most of the radiation emitted by pool fire flames

it is necessary to accurately model soot content in order to accurately predict radiation fluxes from the flames. ARCHES employs a simple, empirical soot model that depends on the stoichiometric fuel mass fraction and the concentrations of certain carbon-containing species. The accuracy of this model is far from certain, so a soot parameter is introduced to the model that can greatly enhance or inhibit the soot content predicted in ARCHES simulations. The uncertainty of this model is such that an order of magnitude in either direction is applied for the range of tested soot parameters.

With these model parameters in mind a series of simulations are run using the 200^3-node resolution determined from the grid convergence tests. Five simulations are run that test the possible range of the input parameters thought to have the highest influence on the model output (i.e., the mass transfer coefficient and the soot model amplification factor.) The conditions and results for each of these tests are shown in Table 6.1. The data from these simulations provide a range of possible value for the mass burn rate and pool surface radiation that can be compared with the experimental data.

The consistency analysis here involves comparison of 10 dataset units, the global, steady mass burn rate, and steady pool surface radiative fluxes at nine different radial locations. The locations include the pool center, 2, 4, 6, 8, 10, 12, 13, and 14 centimeters from the center. The simulations provide pool surface radiation data that is interpolated at eight equally spaced locations for a given radius. The global value for a particular radius is the average of the eight locations. The final value for each radius is a time average over the time period of the simulation that is considered steady state.

The first step in the consistency analysis is to compare the experimental data and their error with the range of values predicted from simulation. Figure 6.22 shows the comparison for the global steady mass burn rate, and Figure 6.23 shows the comparisons for the incident pool surface radiation at the different radial locations. Klassen and Gore (1992) do not report at length in regard to the determination of experimental error for the measured quantities of interest. They state that they estimate the mass burn rate error to be $\pm$ 5% and the radiation measurement error to be $\pm$ 10%. Nakos (2005) performs an error analysis for instruments similar to the Gardon-type radiometer used by Klassen and Gore. For measured radiation fluxes, Nakos finds that for wind driven pool fires, radiation

measurement error can be as high as ± 20%, and the same error for buoyant fires like the ones of interest here can be as high as ± 40%. An estimate of ± 10% for radiation measurement may very well be optimistic. For the purposes of this analysis an error of ±10% is assumed for the mass burn rate, and an error of ±20% is assumed for the radiation measurements, and these errors are reflected in Figures 6.22 and 6.23.

The next type of consistency test is a series of pair-wise analyses. In this series of tests, two dataset units are used to form a simple dataset. The consistency test seeks to find a region in the input parameter space (which consists of the range of values for the mass transfer coefficient and the soot parameter as shown in Table 6.1) that gives model predictions consistent with the experimental values within the assumed error range. The pair-wise tests produce a consistency measure as defined by Feeley et al. (2004). If the value of the consistency measure is greater than zero, then the dataset is said to be consistent, and if the value of the consistency measure is less that zero, then the dataset is said to be inconsistent. For a dataset consisting of a single dataset unit, consistency can be determined from inspection of a plot of the experimental point with its error range and the range of values from simulations. If there is overlap in the range of experimental and simulated values then there is consistency. This is readily seen in Figures 6.22 and 6.23. When a dataset is larger than one dataset unit determining consistency becomes more complex.

In the pair-wise tests each possible combination of two dataset units is used to form different datasets and a consistency measure is taken for each dataset. With 10 available dataset units, there are 55 unique pairings possible if the possibility of pairing a dataset unit with itself is included. Such a series of tests can reveal specific inconsistencies within the larger dataset, and it can reveal where the inconsistencies are happening. If there are any inconsistencies in the pair-wise tests then it is impossible for the entire dataset to be consistent.

Figure 6.24 is a bar graph that shows the consistency measure for each pair-wise test possible with the 10 dataset units involved in this study. The bars are grouped into 10 clusters. The first cluster on the left-hand side of the bar graph represent consistency measures for all pair-wise tests involving the first dataset unit, the second cluster represents

the same quantity for all pair-wise tests involving the second dataset unit and so on. Within the first cluster are 10 individual bars. The first bar in the first cluster (going from left to right) gives the consistency measure for the pair-wise test that involves the first dataset unit with itself. The second bar in the first cluster represents the consistency measure for the pair-wise test involving the first and second dataset units, and so on. With the data represented this way there are redundancies in Figure 6.24. For example, the third bar in the first cluster should have the same value as the first bar in the third cluster because these bars represent consistency measures for pair-wise test involving the first and third dataset units.

Examination of Figure 6.24 shows that there are inconsistencies with some of the pair-wise tests, and all the inconsistencies arise between the mass burn rate (the first dataset unit) and most pool surface radiation fluxes. This confirms previously stated concerns about the radiation predictions from ARCHES. A subsequent consistency analysis using all 10 dataset units confirms that the whole dataset is indeed inconsistent as expected. In addition to the consistency measure, a sensitivity analysis can reveal which components of the dataset are most likely responsible for the inconsistency of the dataset. Figure 6.25 shows the Lagrangian multipliers for each part of the dataset. The larger the magnitude a Lagrangian multiplier has the more sensitive the consistency measure is to the quantity with which the multiplier is associated. The quantities that are assigned Lagrangian multipliers are the upper and lower bounds of the experimental data and the model inputs. The heptane pool system features 10 data points and two model parameters. Each of these has an upper and lower uncertainty bound. This creates 24 Lagrangian multipliers. Figure 6.25 shows that one multiplier has a value orders of magnitude larger than the others. This multiplier (the fifth multiplier) is associated with the lower bound in the error associated with the measurement of the mass burn rate. This could indicate an issue with the measurement of this quantity.

For justification purposes with the Responsive Boundary model, there is a good region of consistency for the mass burn rate alone. However, since the burn rate is so closely correlated with the incident radiative heat flux, a consistency analysis involving the mass burn rate and the pool radiation measurements makes a more meaningful analysis for the

purposes of determining the physics within the heptane pool fire system.

The issue at hand is that individually there is consistency for the mass burn rate alone, and there is consistency for all the radiation measurements taken collectively, but for some reason when coupled together, there is an inconsistency issue. This inconsistency can be understood through examination of the comparisons in Figures 6.22 and 6.23. All the data comparisons in these figures show strong overlap between experiment and simulation. However, there is a differential in the specific region of the overlap between the mass burn rate and the radiation data. In the case of the mass flux the experimental range overlaps the simulation range at its center. In the case of each of the radiation flux comparisons the experimental range overlaps the simulation's range at its lower end. The regions of the simulation space are comparable between consideration of mass flux and radiative flux because these two properties are correlated (i.e., when radiative heat flux increases, mass flux also increases.)

The model predicts the mass burn rate very closely compared to the experimental data, but it over-predicts the radiation values. There are at least three possible explanations for the inconsistency: (1) the Responsive Boundary model is under-predicting the mass burn rate for the given radiation flux to the pool surface, (2) ARCHES is over-predicting the radiation due to modeling error with the empirical soot model or some other model internal to the code, and (3) the radiation flux measurements offered by Klassen and Gore (1992) have an experimental measurement bias that under-predicts the radiation flux values.

In other words, the problem is in the models, or it is in the experimental data, or some combination of both. Neither explanation (1) or (2) alone can resolve the inconsistency. Most of the likely causes of error in the Responsive Boundary model have been accounted for. The results from the simulations summarized in Table 6.1 indicate that the mass burn rate and radiation values are very sensitive to the soot factor and show minor to negligible responses to the range of mass transfer coefficients, and the mass transfer coefficient is considered to be the most likely source of error from the Responsive Boundary model. Next, if ARCHES were to be adjusted to give radiation values more consistent with experiment through its soot model or by some other means, the Responsive Boundary would simply give lower mass burn rates in response, and the inconsistency would still exist between the

mass burn rate and the radiation measurements. In contrast, if the radiation measurements turned out to have a systematic bias toward under-prediction, that alone would explain the inconsistency in the current dataset. Examination of the radiation value comparisons in Figure 6.23 show that the values from experiment and simulation both show a similar trend with respect to pool radius. In both cases the radiative flux is highest near the pools center, and it gradually weakens toward the edge of the pool. The main difference between the experimental and simulated fluxes is that the simulated values are higher than the measurements by similar percentages. This similarity in trend supports the possibility of a systematic bias in the radiation measurements.

Table 6.2 shows a graph of results from additional consistency analyses performed on the full dataset. In this series of tests the assumed experimental error in the mass burn rate is varied from 10% to 20%, and the assumed error in the radiation measurements is varied from 20% to 40%. The radiation errors are shown in the vertical axis and the errors in the mass burn rate are shown along the horizontal axis. The intersection of each value for these errors represents a consistency analysis test. This makes for a total of 25 tests to determine a range of possible experimental errors for both pool surface radiation values and mass burn rate. The results of each consistency test are shown at the intersections of the different error values. An 'X' symbol indicates an inconsistent test result. The '$\surd$' symbol indicates a consistent result.

The data in Table 6.2 suggest that in order to find a region of consistency, it is necessary to have an error of 40% in the upper bound of radiation measurements and a minimal error of 15% in the lower bound of the mass burn rate measurement.

Since the authors of the pool fire study did not include a detailed analysis of the experimental error it is difficult to say what levels of assumed experimental error are justified. There is some evidence to justify the conclusion of a bias error in the radiation flux measurements to the pool surface, but the purpose of data collaboration is to bring computationalist and experimentalist together to arrive at a consensus as to where the truth most likely lies. More input is certainly needed from experimentalists as to the error in the radiometers before the assumption of measurement bias can be justified.

Subsequent discussions with Nakos (2011) provide more information on the Gardon-type

radiometers that Klassen and Gore (1992) use in their measurements. Finding agreement between experiment and simulation is generally a difficult task with heat flux measurements. For example, the Gardon gages have proven to be unreliable in mixed radiation and convection environments like those seen in pool fires. Gardon gages obtain flux readings from a parabolic temperature profile on the surface of their sensing element. Convective shear flows affect the temperature profile sufficiently to create error in the callibration constants that are obtained from callibration procedures that use only radiative sources.

Another potential source of error for the Gardon gages is the possibility of condensaton of water or fuel products on the sensing surface if the gage temperature is below the dew point. This is more common with radiometers that are cooled. Klassen and Gore do apply tap water to the gages with which they measure the heptane fires, but they also apply a nitrogen purge to prevent condensation. Despite this counter-measure they do note that condensation occurs with their toluene experiments which prompts them to use warm water as coolant instead of normal tap water for toluene. It is possible that despite the measures they apply to prevent it there may have been some condensation with the heptane fires as well. The deposition of soot and other combustion products tends to occur more intensely near the edges of the pool. The increased discrepency between the mass flux data and the radiation data closer to the pool's edge (as seen in Figure 6.24) supports this trend. In addition to the possibility of surface condensation, pool fires in the turbulent flow regime (usually about one meter diameter) experience disturbances in the pool surface. Although the pool fire of interest is smaller than the accepted diameter for turbulent regime, there may be a possibility that some splashing of liquid fuel occurs and floods the meter's sensing surface. If this does occur the radiation flux readings would be lower than they should be. Unfortunately it is not possible to confirm or eliminate either of these possibilities as the authors do not provide more heat flux data.

Another suggestion from Nakos (2011) is to focus the consistency analysis on flux measurements near the center of the pool. The calibration process for heat flux gages involve the use of a blackbody source that fills the viewing angle of the gage. Klassen and Gore use this method, and the gages they use have a very small viewing angle. Gages that are calibrated in this manner experience additional error in their readings when placed in

positions where the heat source does not fill the viewing angle like near the edge of the pool. This implies that heat flux readings closer to the center of the pool are more accurate than those closer to the edge. This observation agrees with the consistency analysis results with the heptane pool fire system. Again, examination of the pairwise test results in Figure 6.24 shows that the inconsistencies between mass flux and pool surface radiation flux (which are shown in the first bars of the all the bar clusters except the first on the left-hand side) are strongest with the radiation measurements nearest the edge of the pool. The mass flux actually shares a region of consistency with the radiation measurement at the pool center, and the value of the consistency measure for the measurements closer to the center are not as negative as for the four measurements closest to the edge.

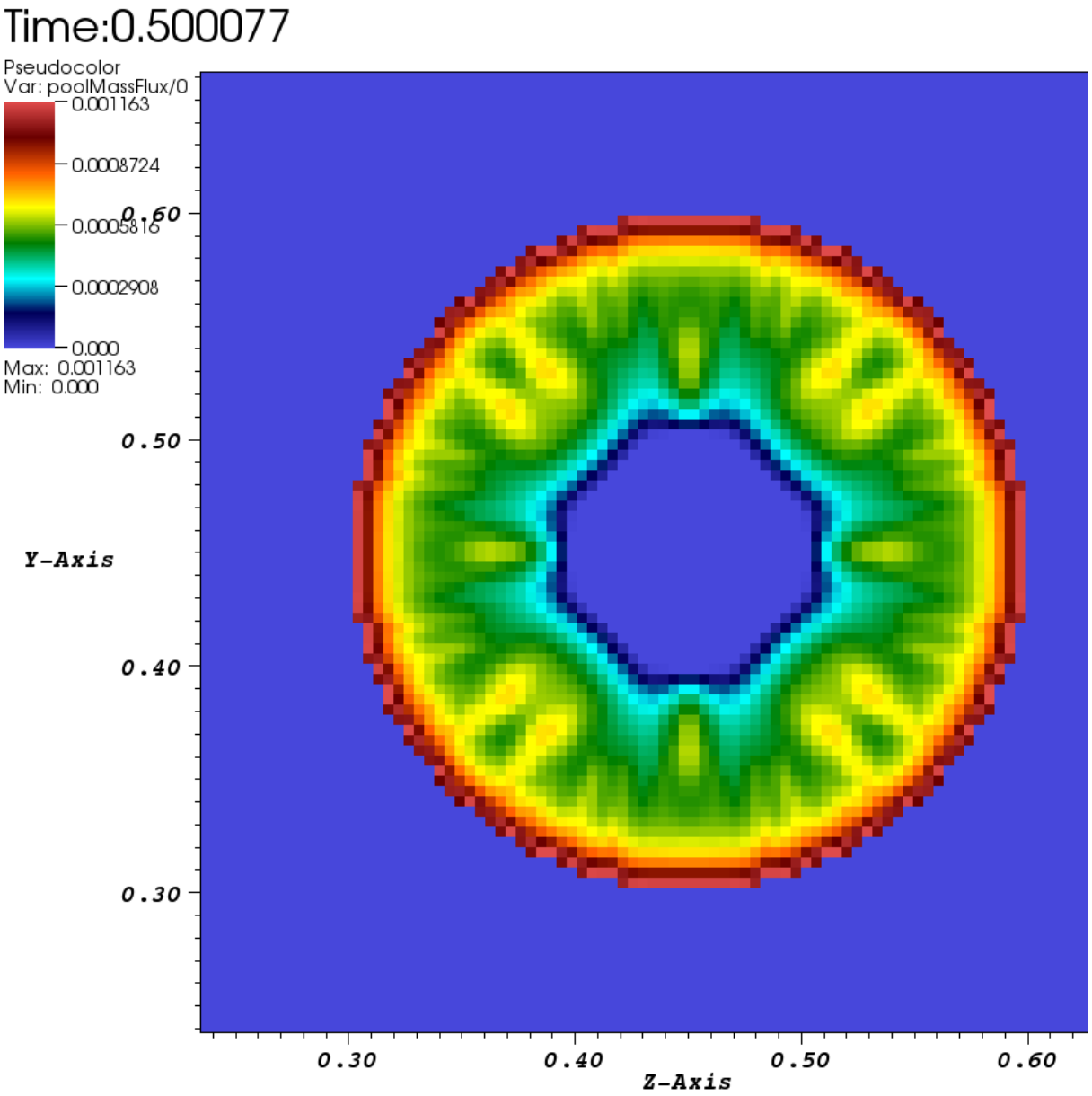

Figure 6.1: The mass flux from the pool surface at a simulation time of 0.5 seconds. This time is during the transient warm up phase of the pool fire.

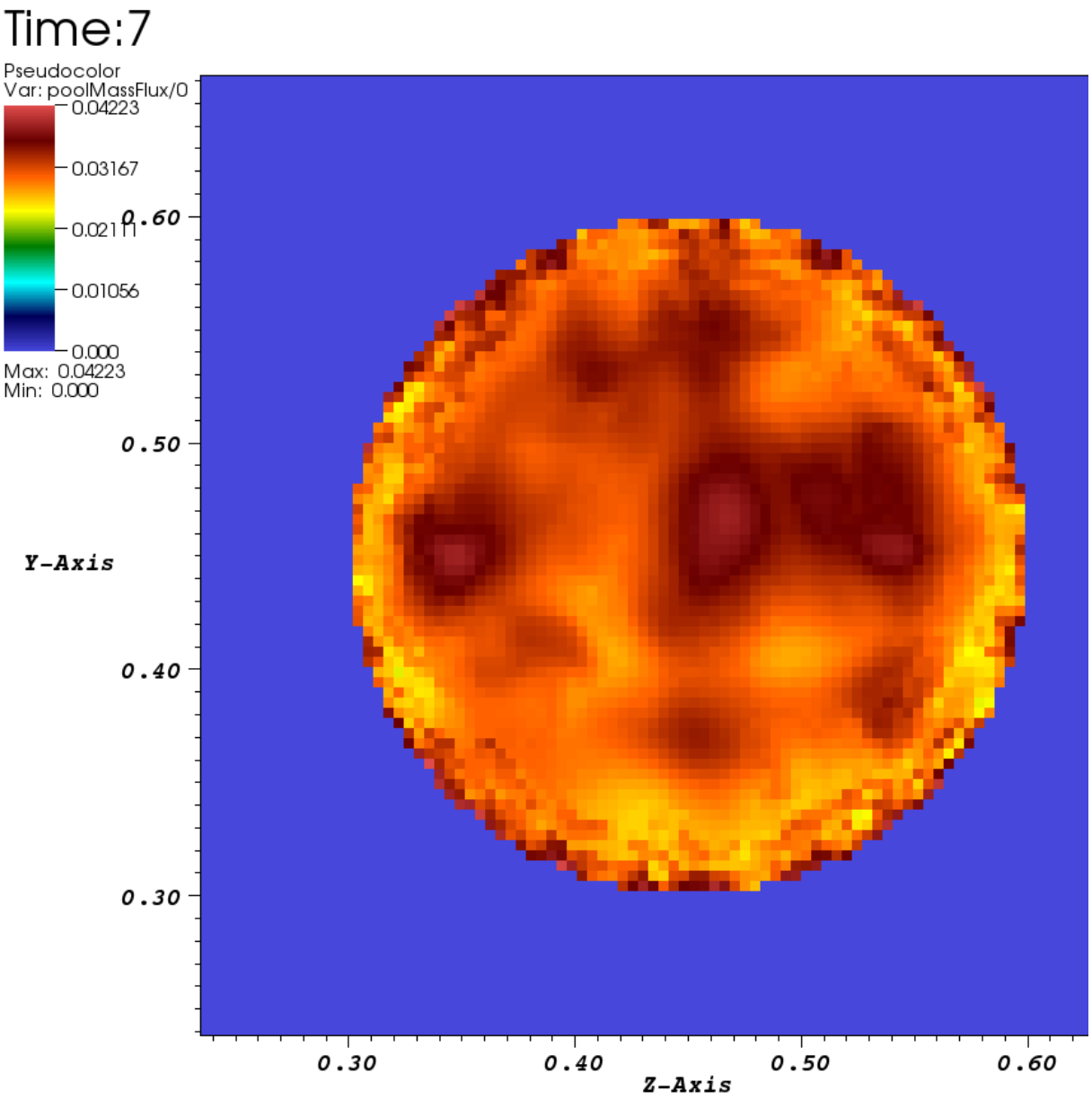

Figure 6.2: The same type of data as that shown in Figure 6.1 at a simulation time of 7.0 seconds. This time is considered to be during the steady burn phase of the pool fire.

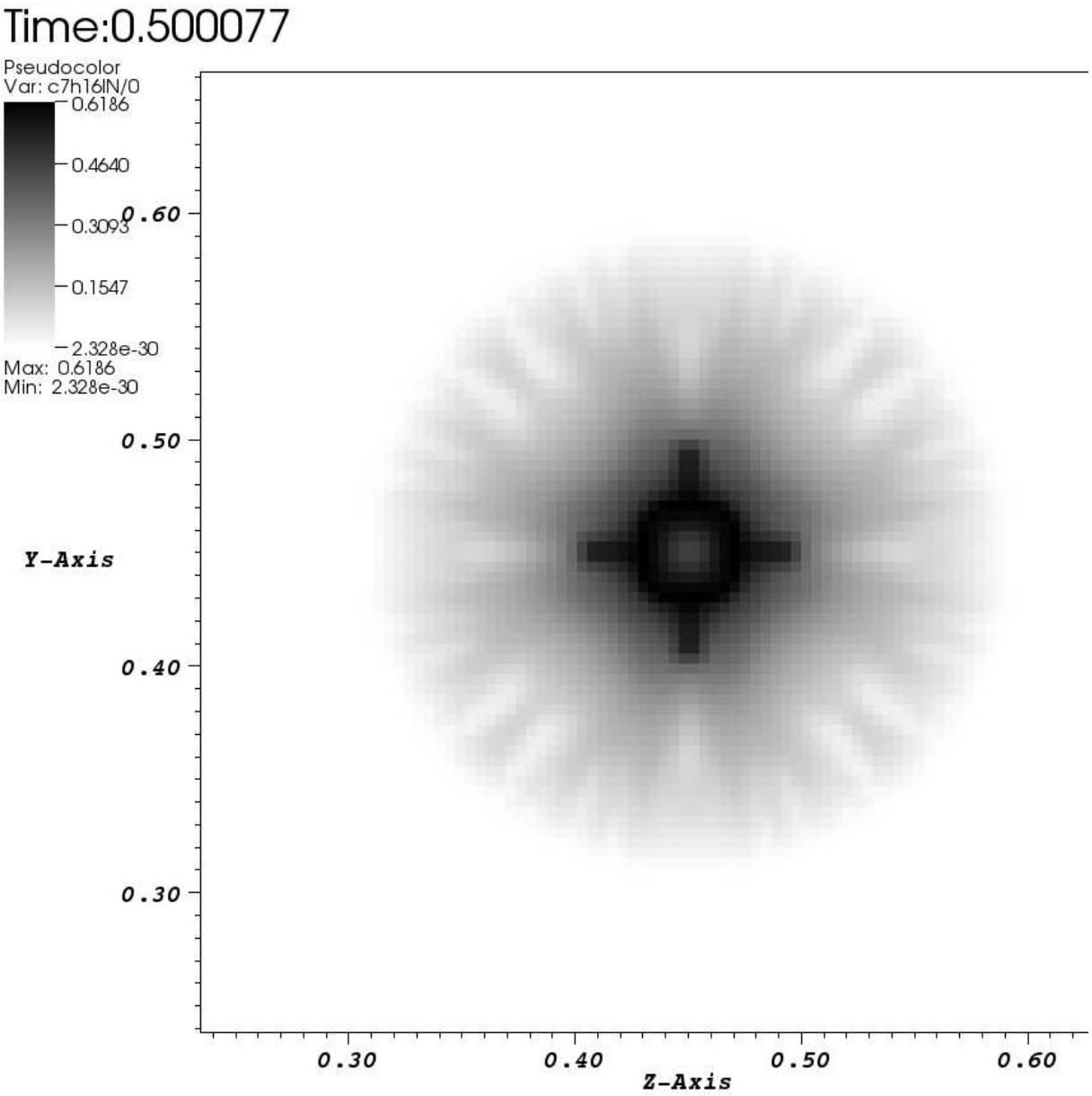

Figure 6.3: The mass fraction of heptane (the fuel) in the gas phase just over (about 2.25 millimeters) the pool surface during the transient burning phase.

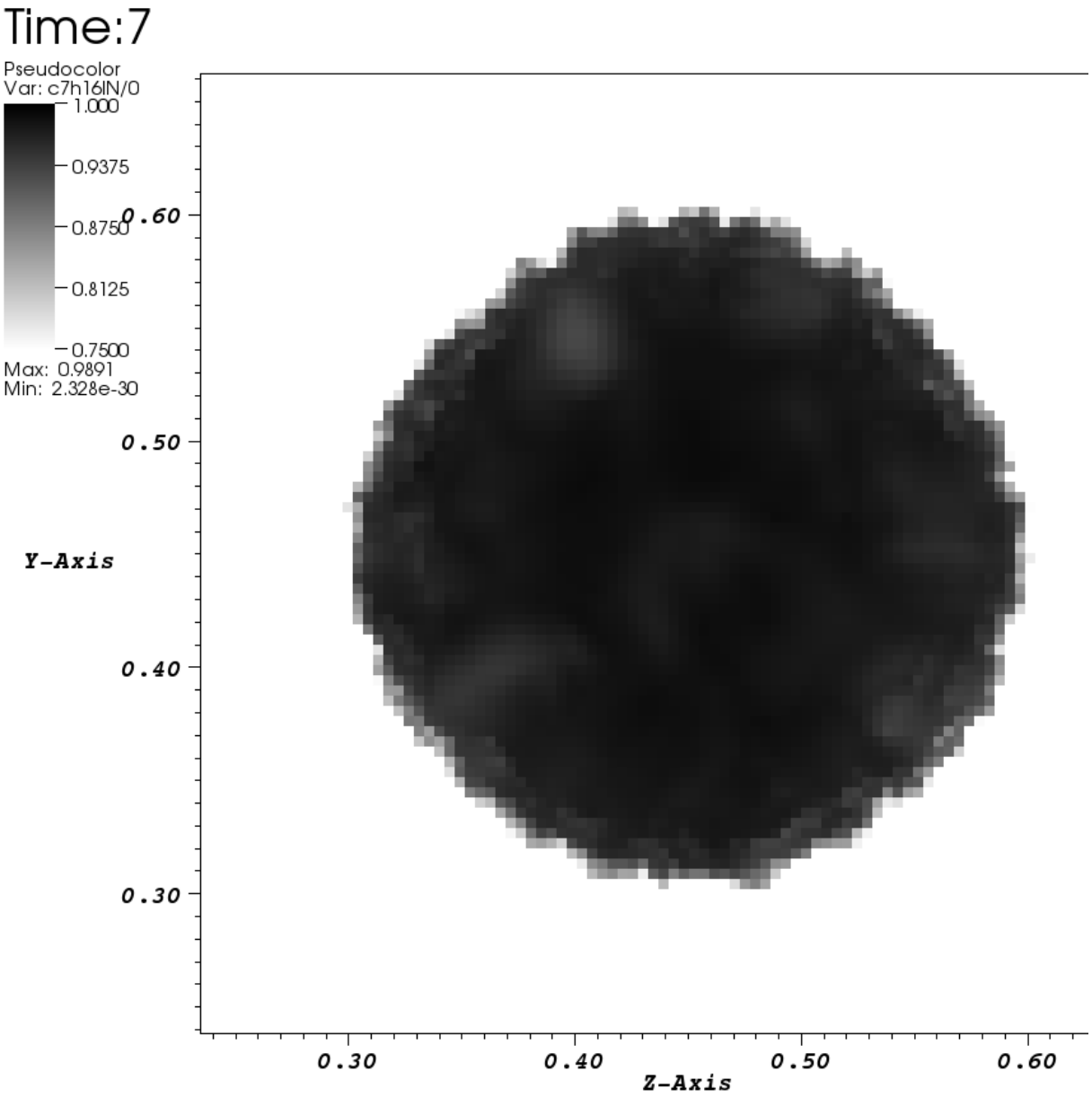

Figure 6.4: The mass fraction of heptane (the fuel) in the gas phase just over (2.25 millimeters) the pool surface during the steady burning phase.

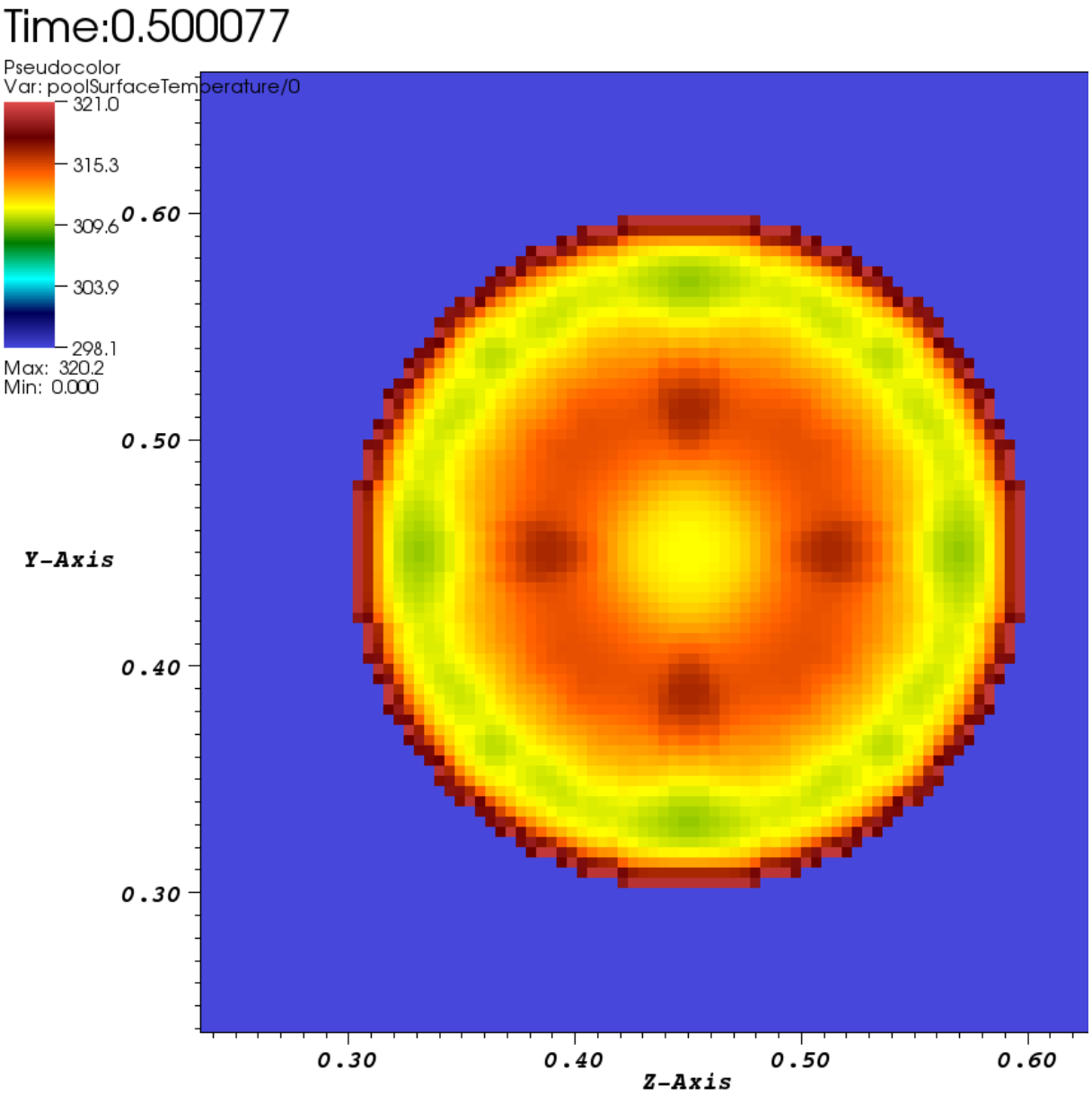

Figure 6.5: The liquid surface temperature of the pool during the transient burning stage.

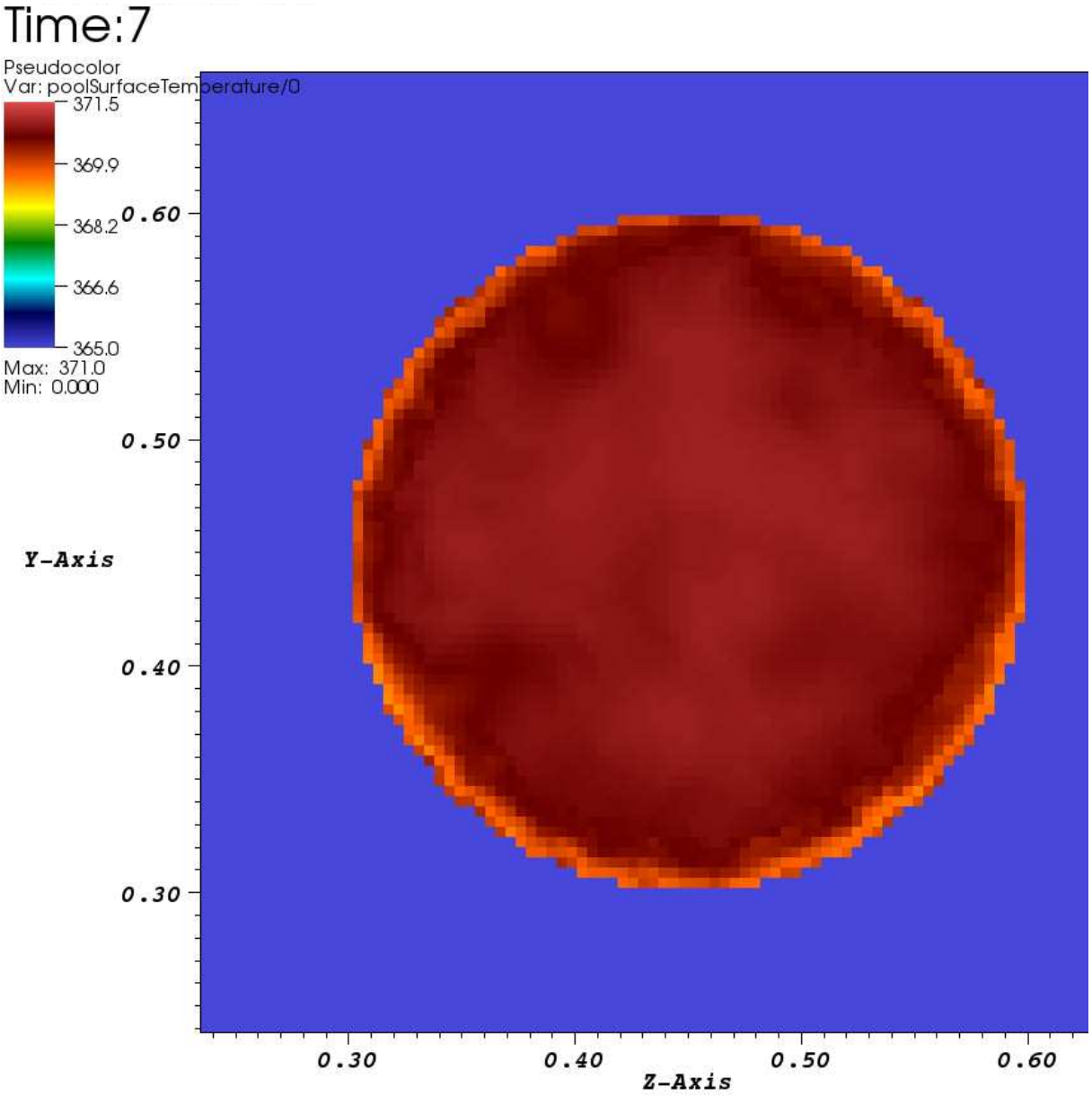

Figure 6.6: The liquid surface temperature of the pool during the steady burning stage.

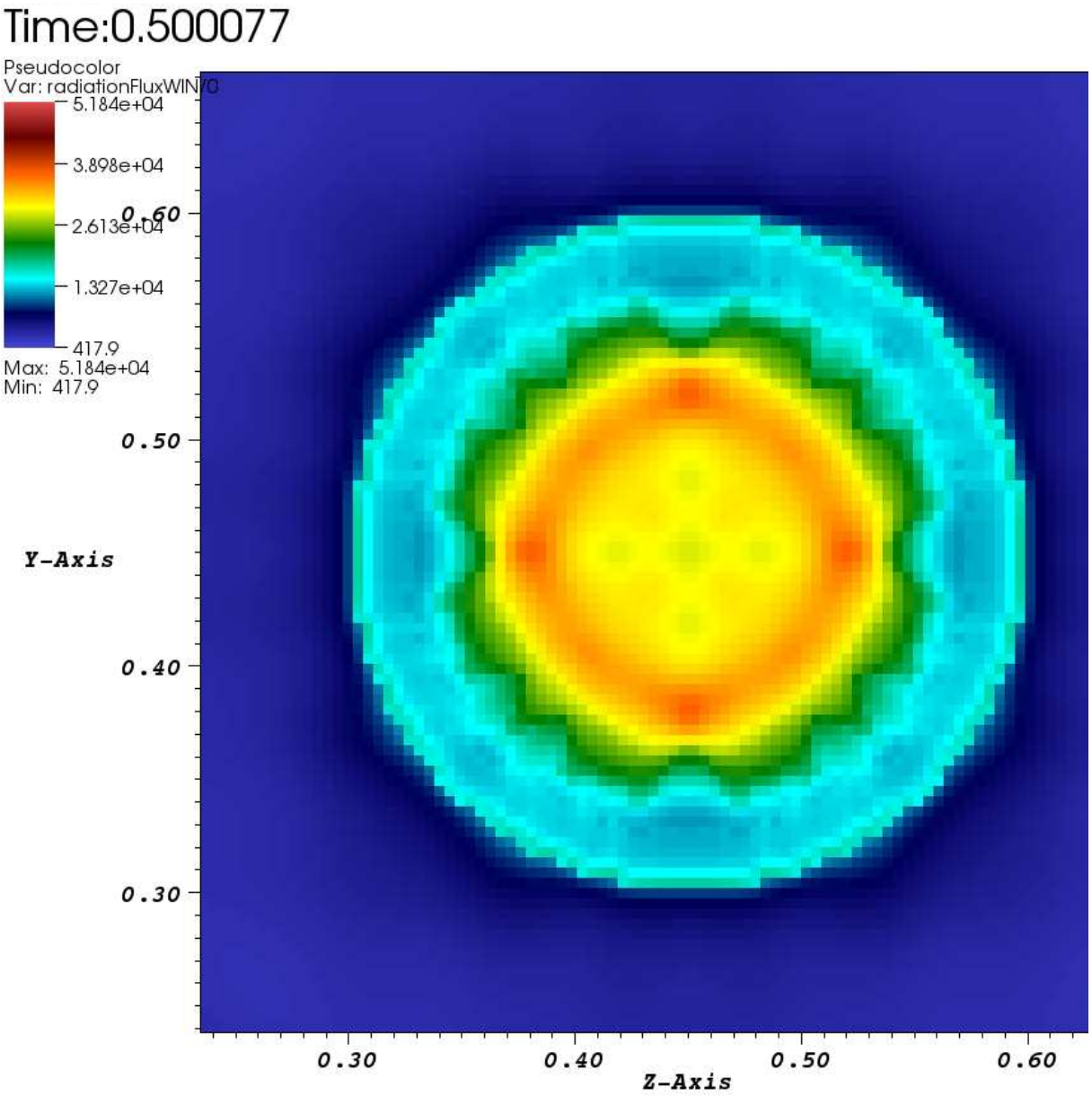

Figure 6.7: The incident radiative heat flux to the pool surface for the transient phase of the pool fire.

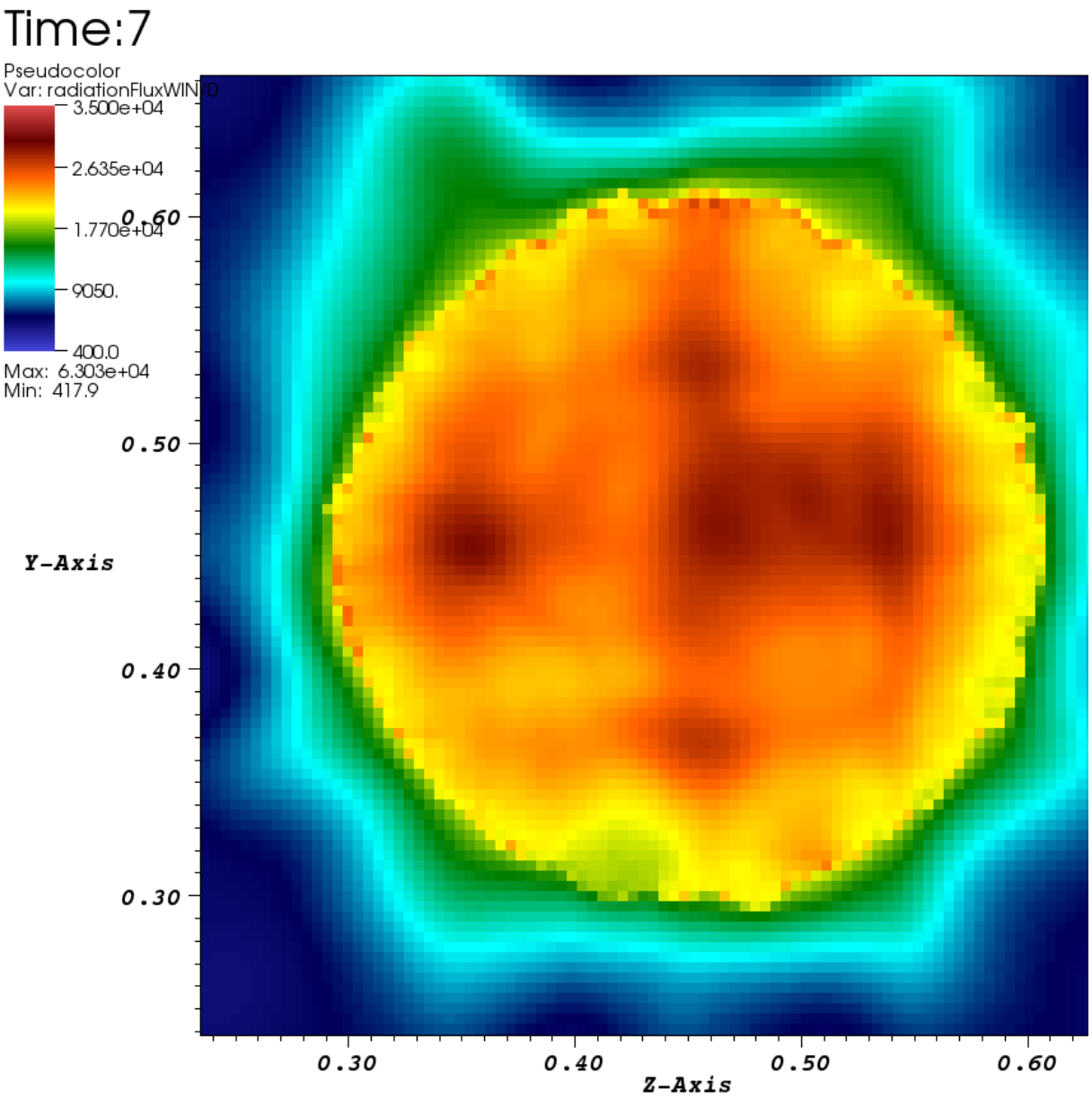

Figure 6.8: The incident radiative heat flux to the pool surface for the steady phase of the pool fire.

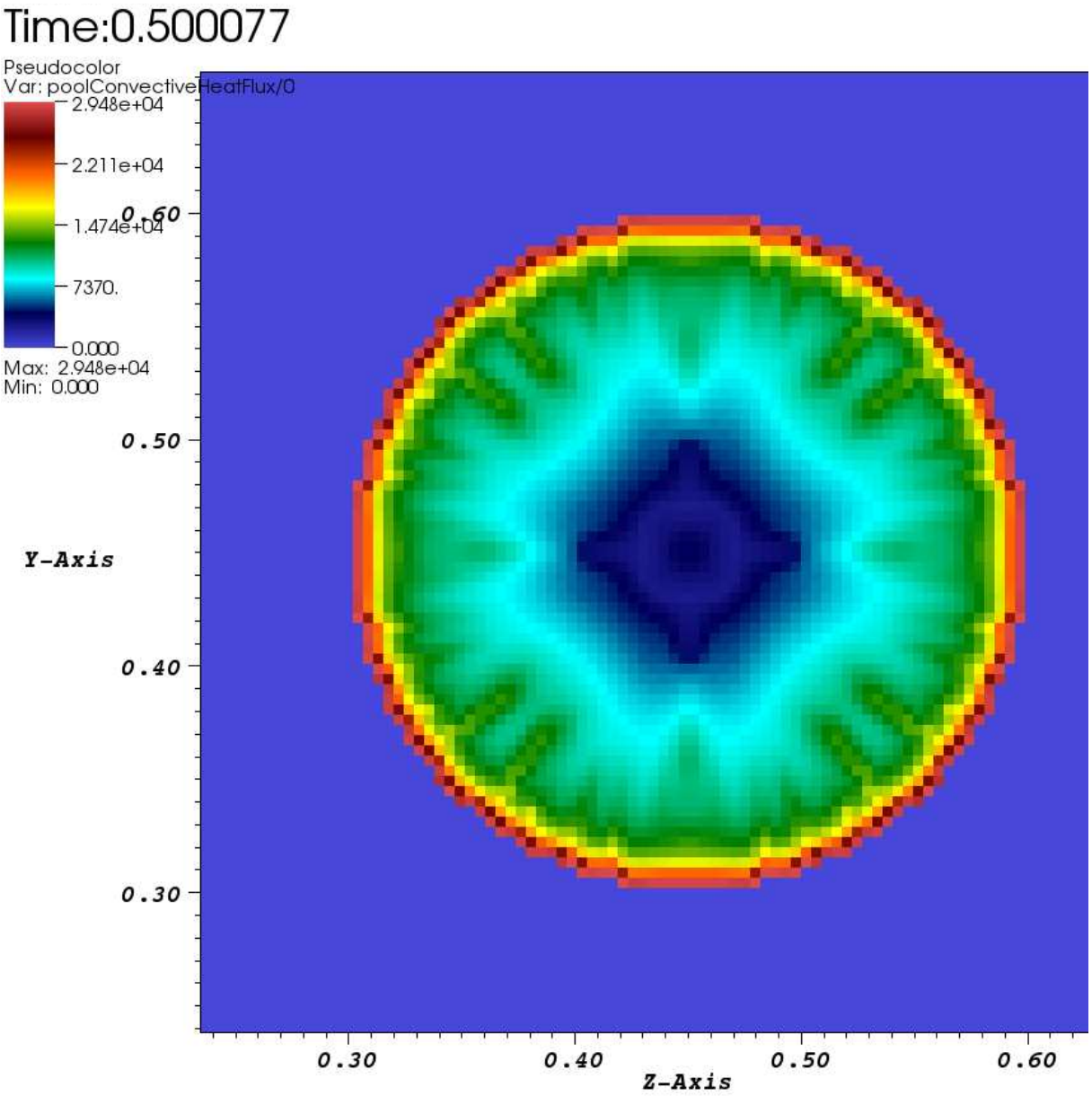

Figure 6.9: The convective heat flux to the pool surface at the transient phase of the pool fire.

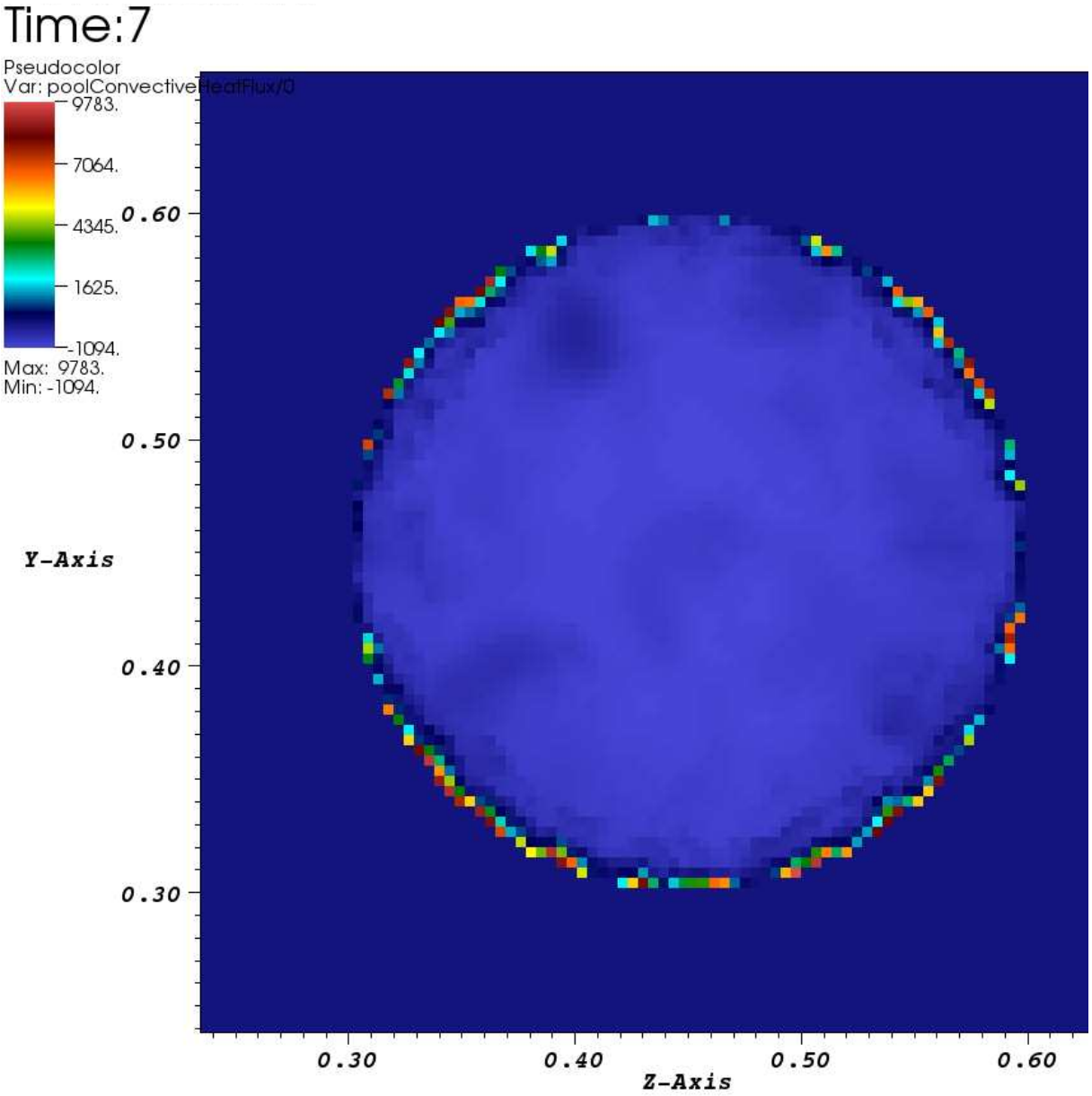

Figure 6.10: The convective heat flux to the pool surface at the steady phase of the pool fire.

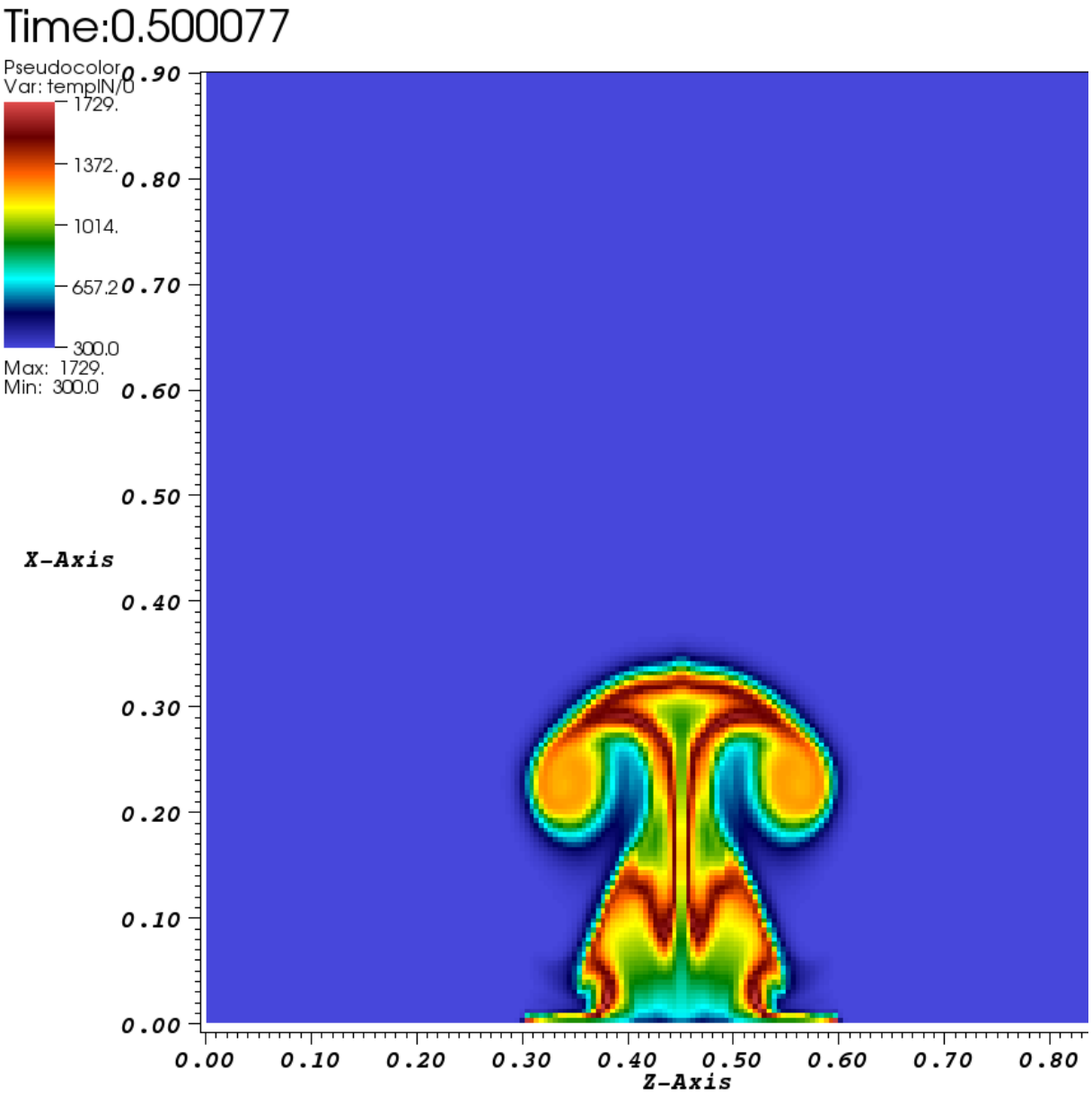

Figure 6.11: A two-dimensional side view of the pool fire. The data are the gas phase temperature profiles of the pool fire during the transient stage.

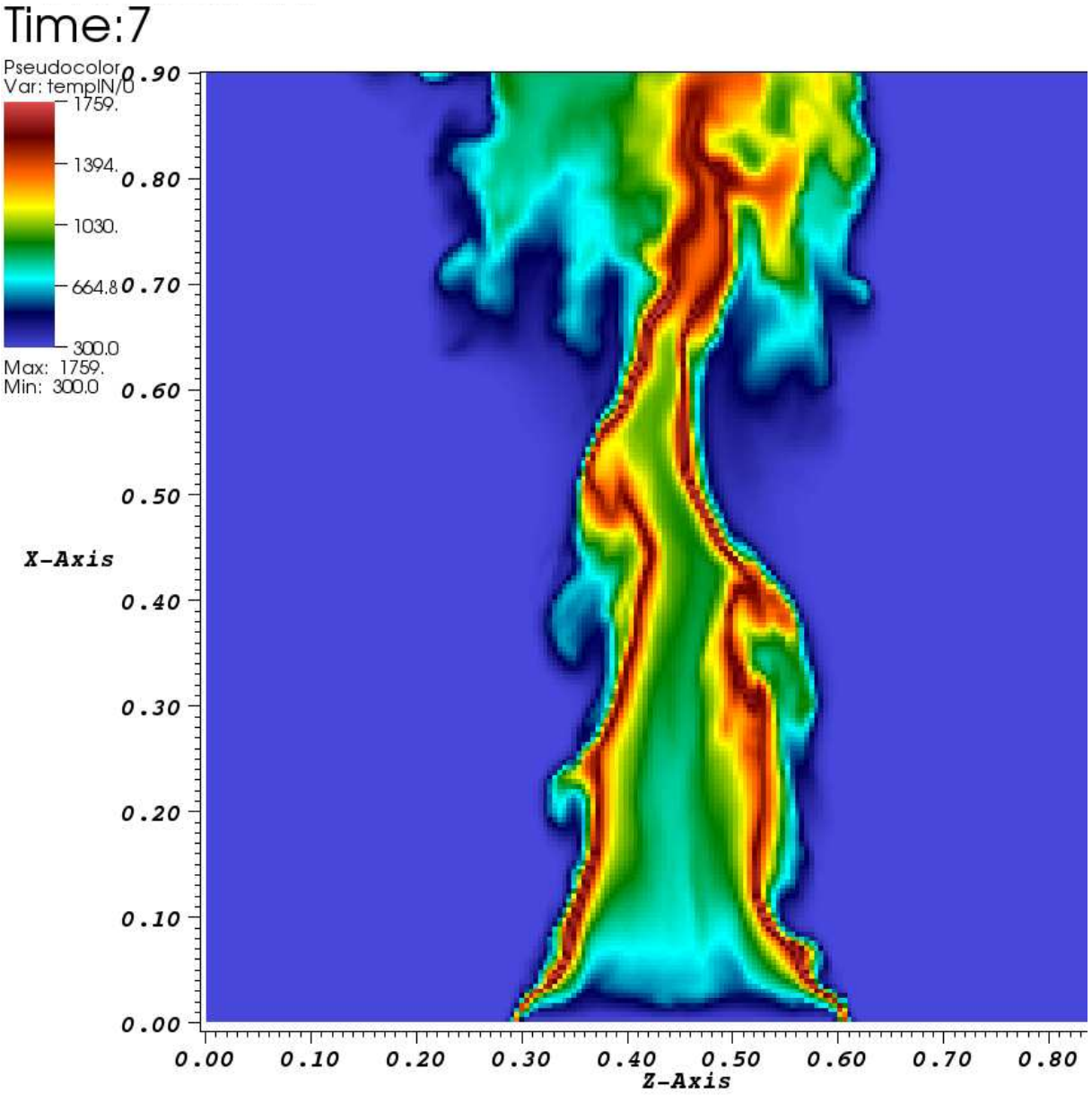

Figure 6.12: A two-dimensional side view of the pool fire. The data are the gas phase temperature profiles of the pool fire during the steady stage.

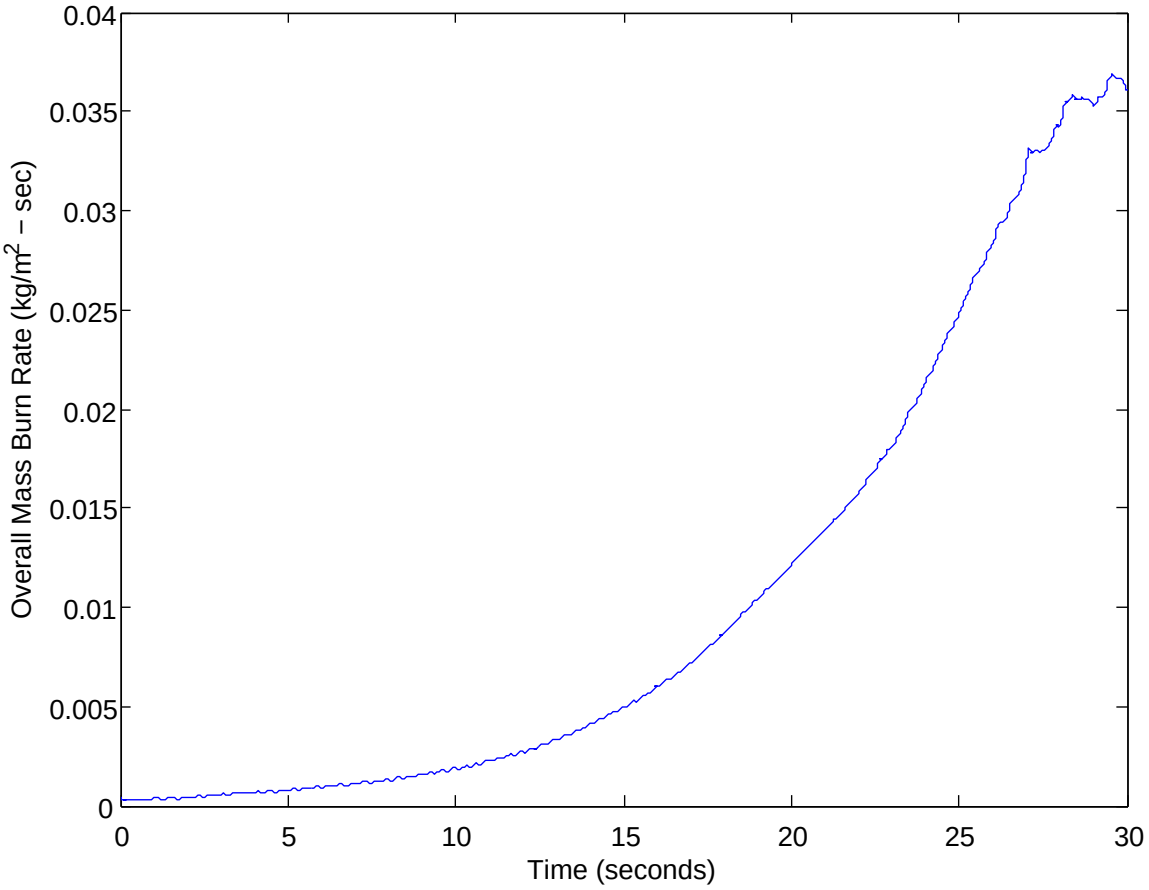

Figure 6.13: A plot of global mass flux from the pool surface with time. As of 30 seconds of simulation time steady state is not yet obtained.

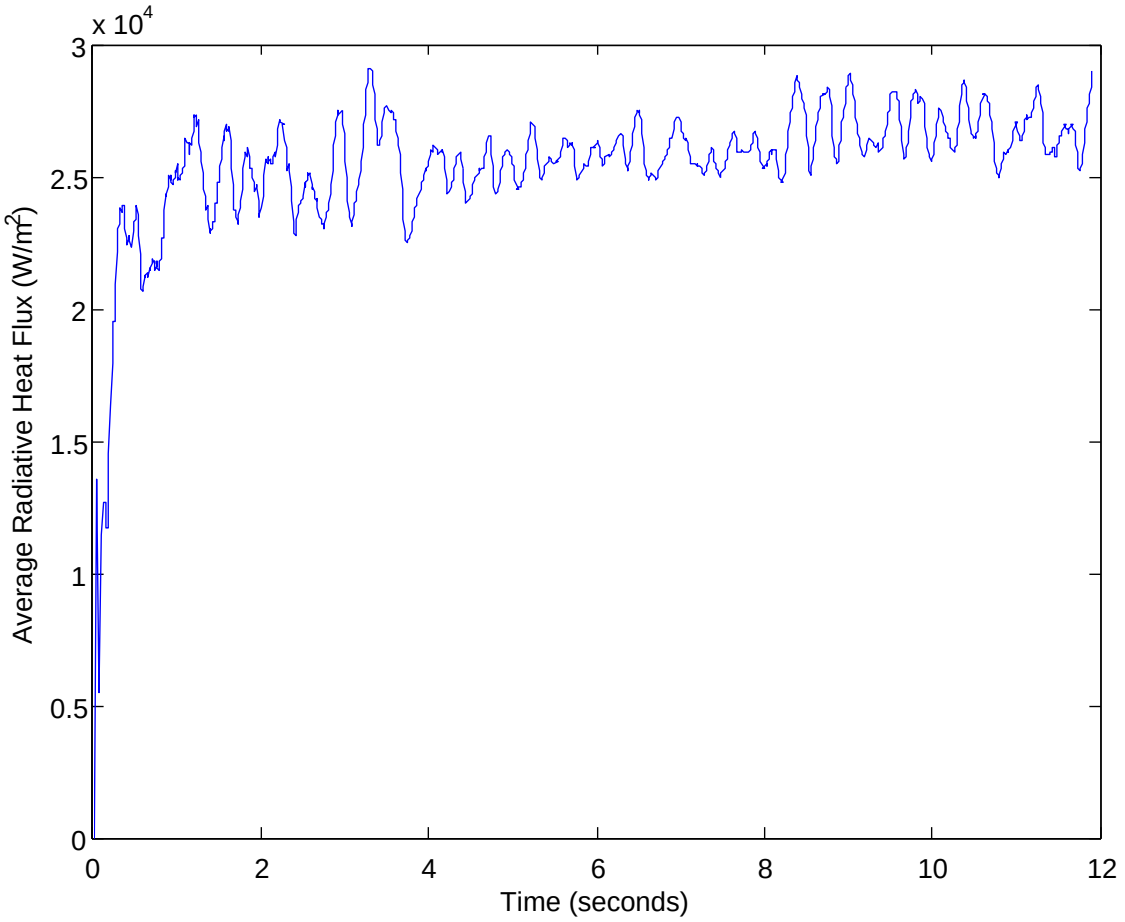

Figure 6.14: The spatially averaged radiative heat flux to the pool surface against simulation time. These data arise from the same simulation used for the data in Figure 6.13.

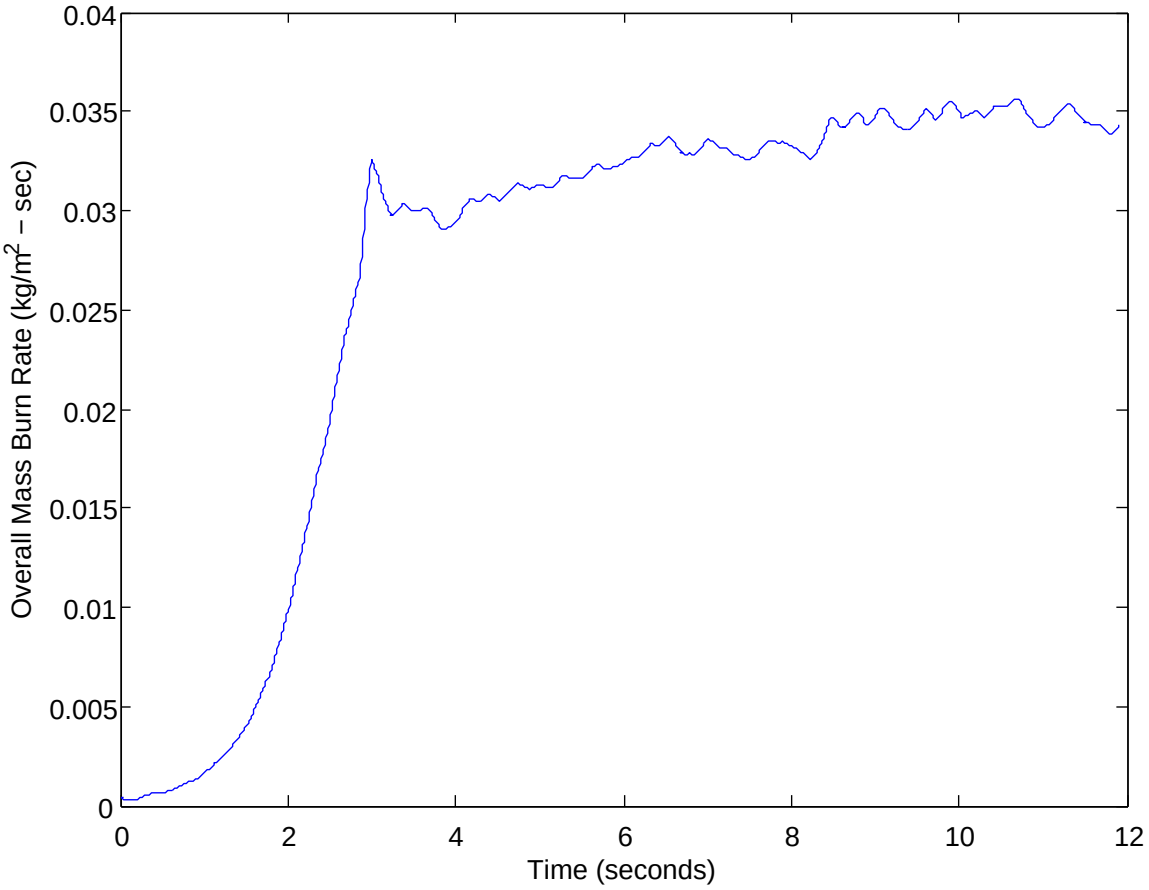

Figure 6.15: The global mass flux for another heptane simulation like the one shown in Figure 6.13. However, this simulation accelerates the pool's response by a factor of 10 for the first 3 seconds of simulation time.

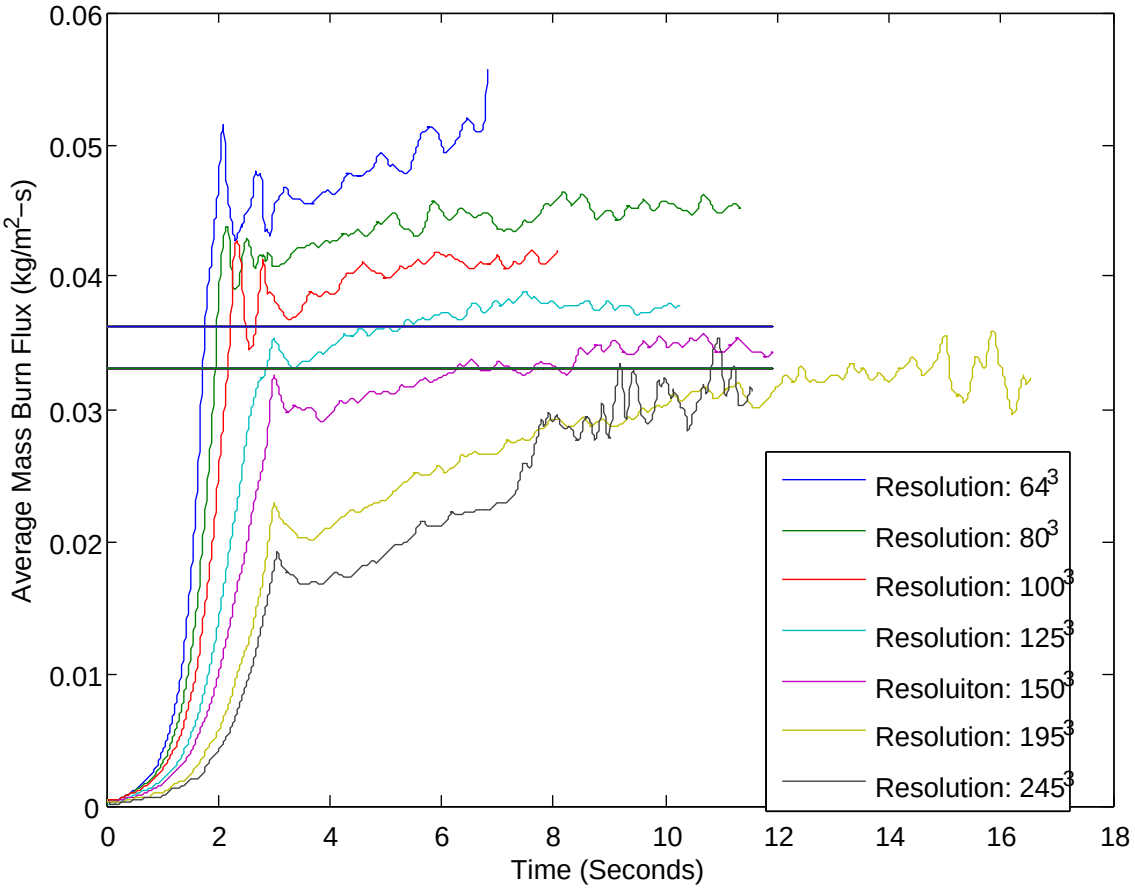

Figure 6.16: The global mass burn flux from the pool surface for several heptane pool simulations at different spatial resolutions.

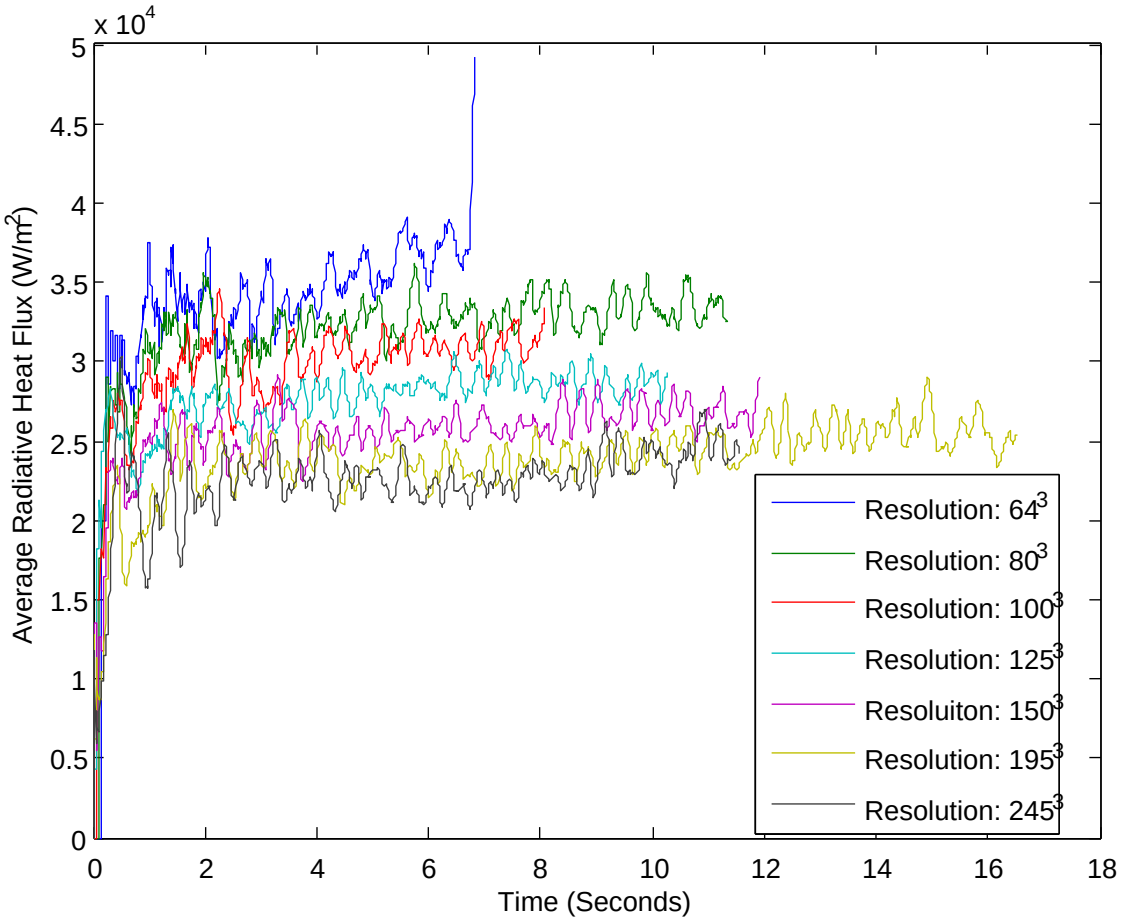

Figure 6.17: The global radiative heat flux to the pool surface for several heptane pool simulations at different spatial resolutions.

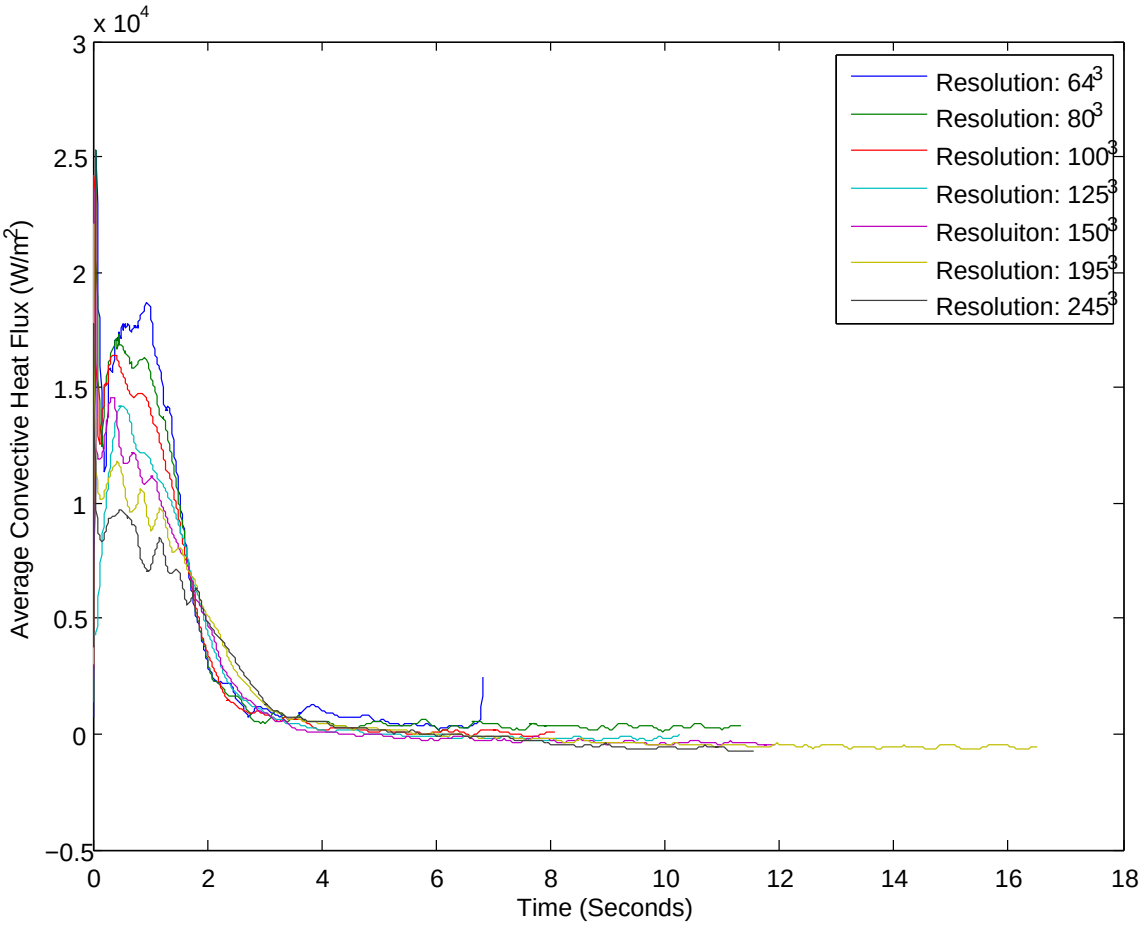

Figure 6.18: The global convective heat flux to the pool surface for several heptane pool simulations at different spatial resolutions.

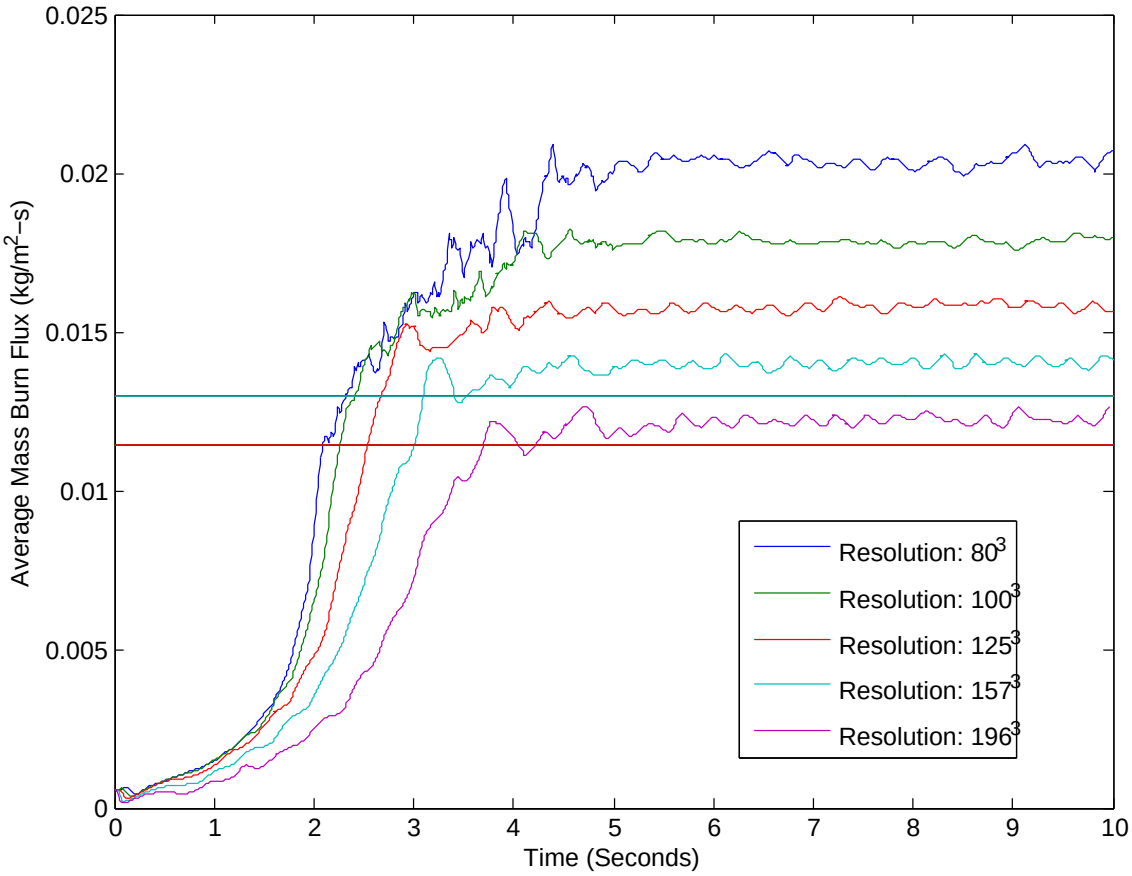

Figure 6.19: The global mass burn flux from the pool surface for several methanol pool simulations at different spatial resolutions.

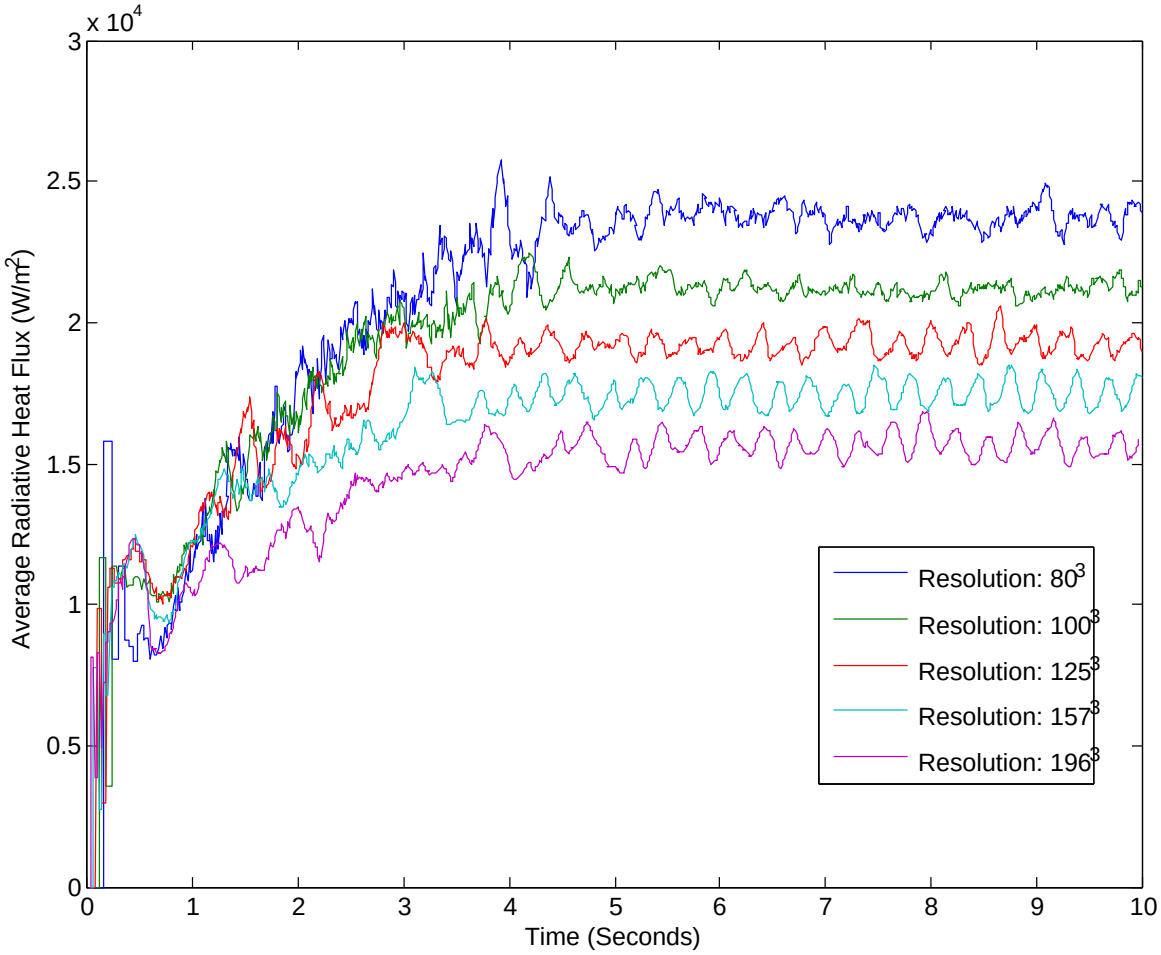

Figure 6.20: The global radiative heat flux to the pool surface for several methanol pool simulations at different spatial resolutions.

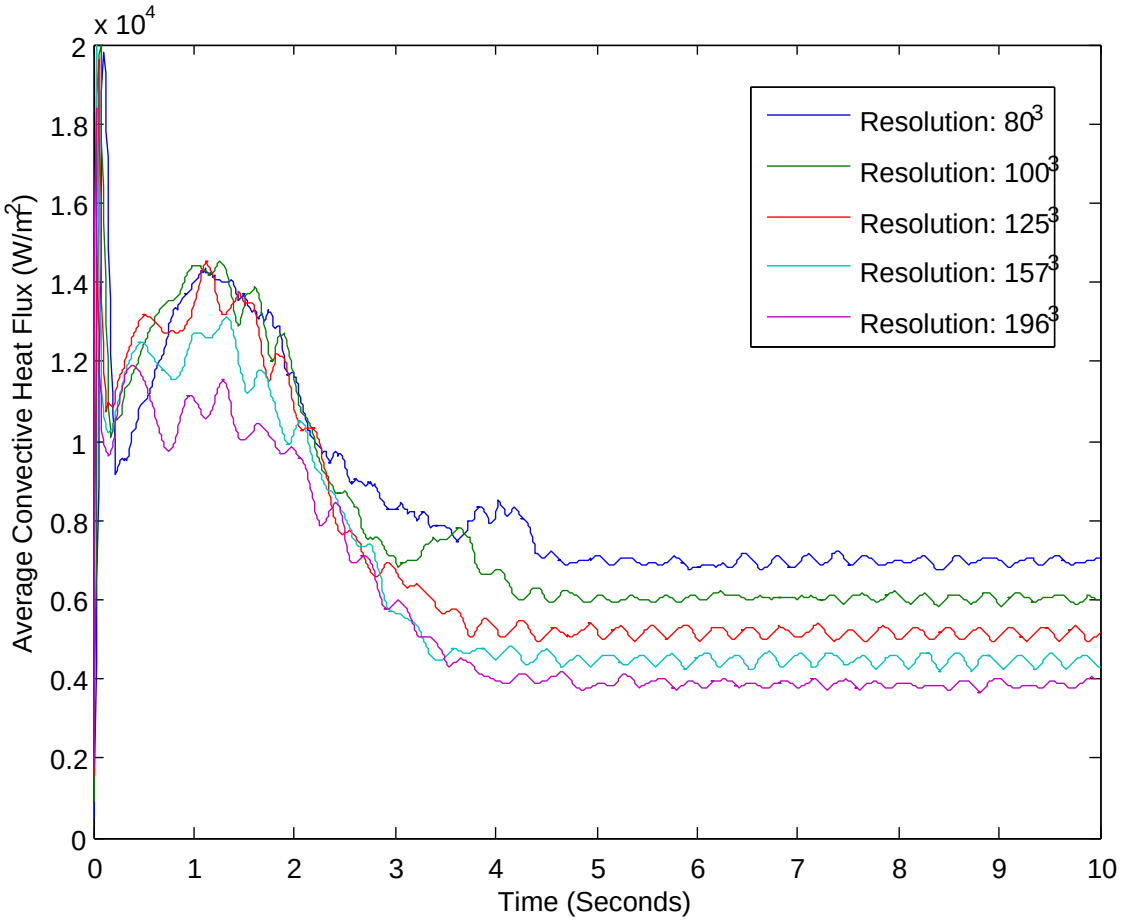

Figure 6.21: The global convective heat flux to the pool surface for several methanol pool simulations at different spatial resolutions.

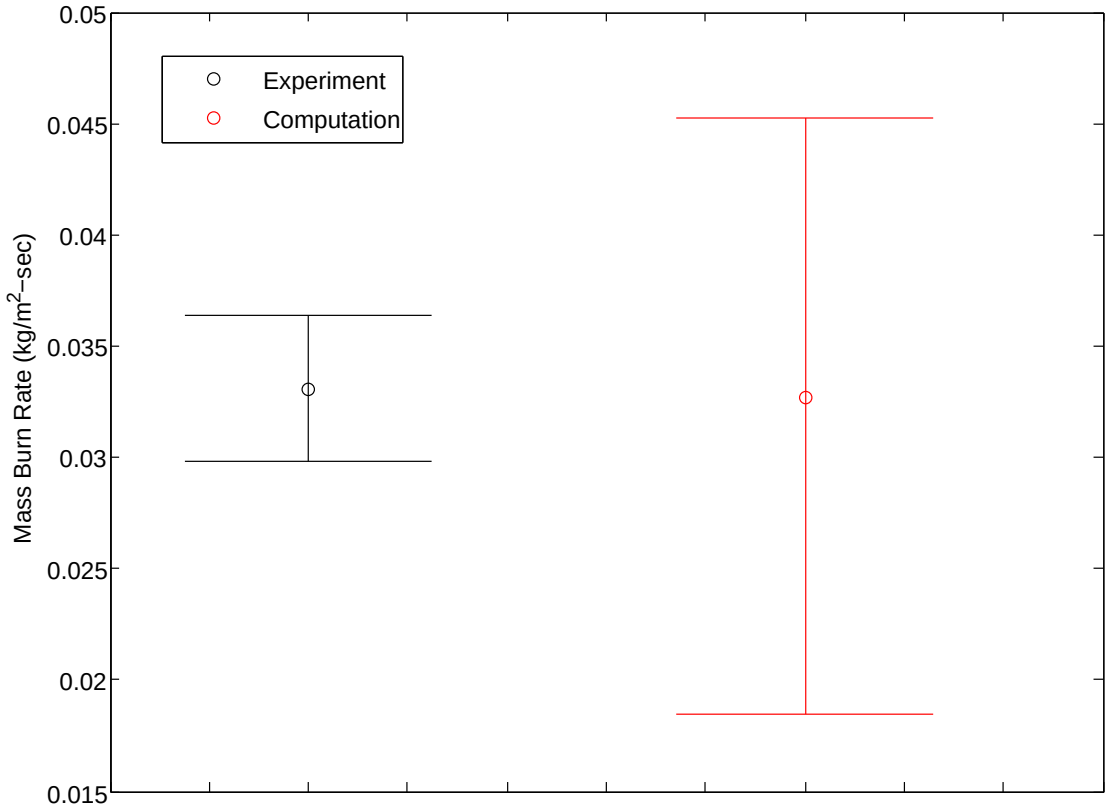

Figure 6.22: The simulation and experimental data with their ranges for the global, steady mass burn flux.

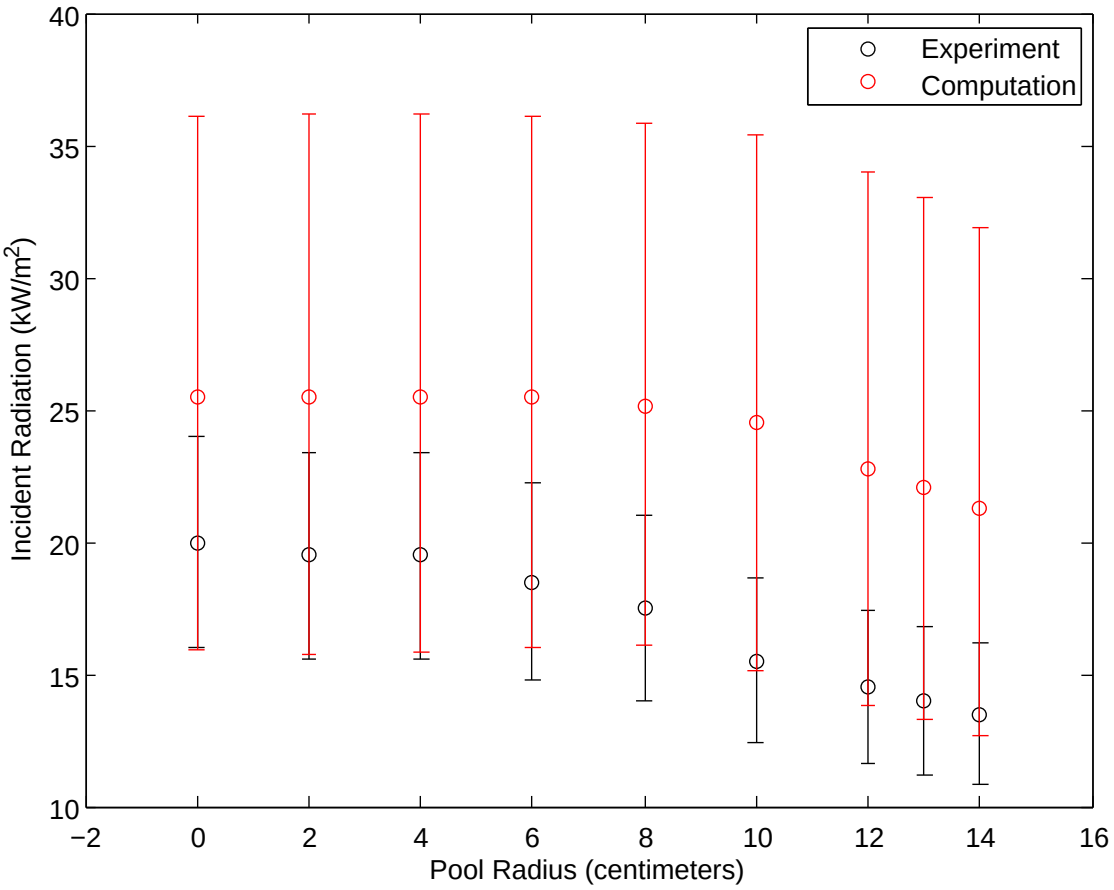

Figure 6.23: The simulation and experimental data with their ranges for the different radiative heat fluxes. Note that a pool radius of zero centimeters denotes the center of the pool.

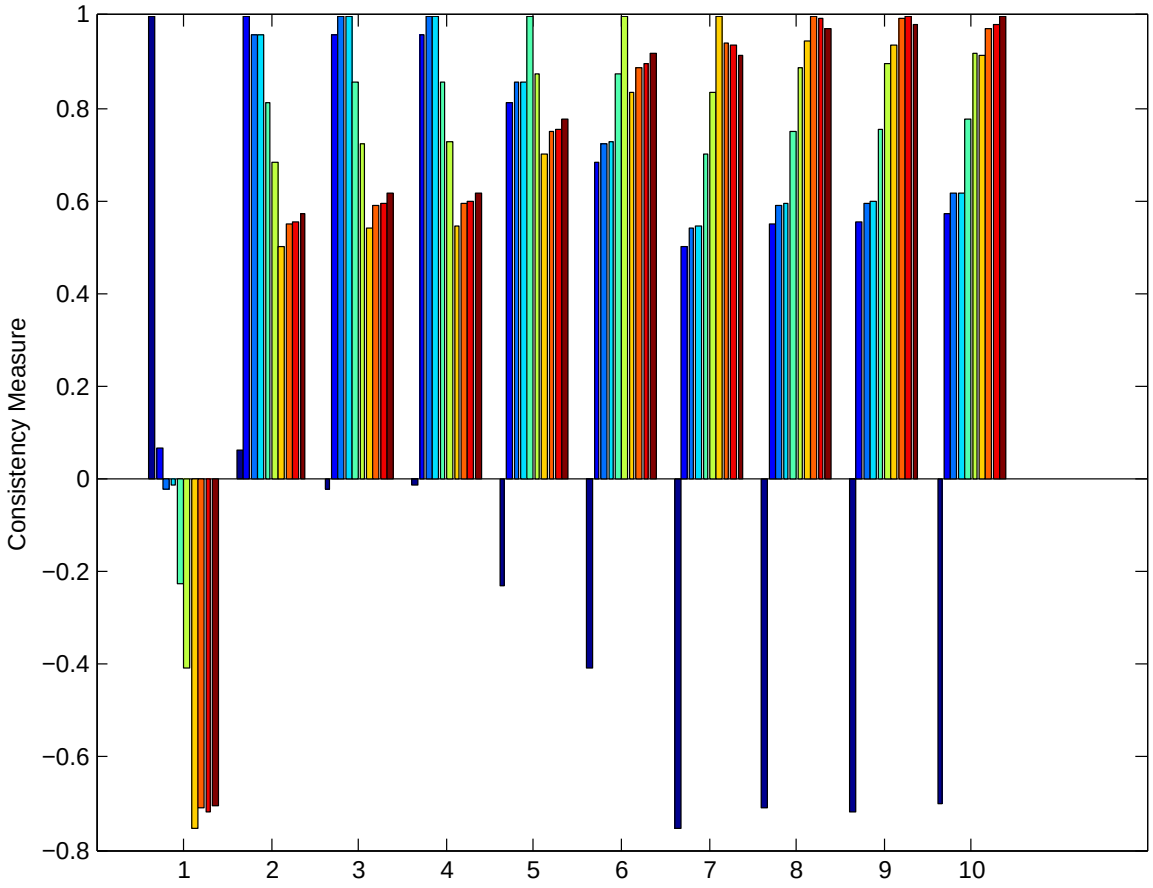

Figure 6.24: Consistency measures for all the pair-wise consistency tests associated with the heptane pool fire dataset.

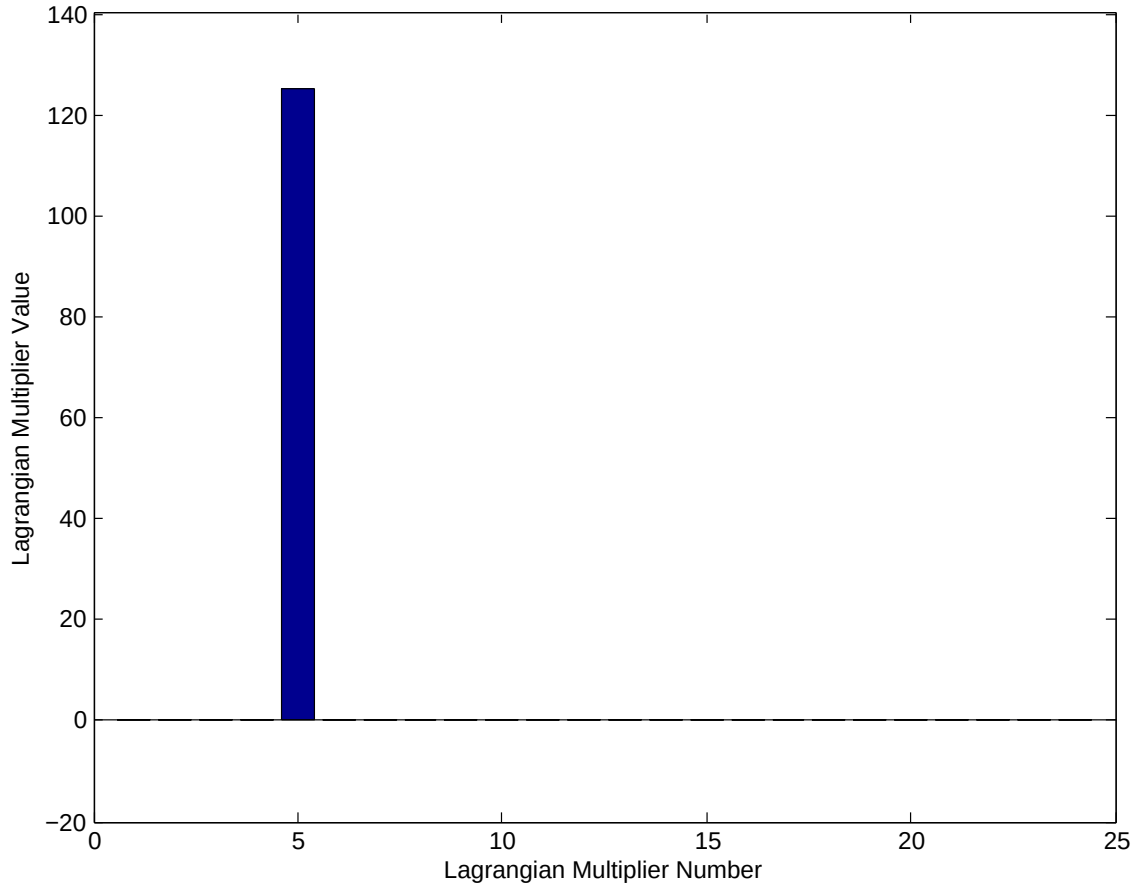

Figure 6.25: The values of each Lagrangian multiplier for each component of the full consistency analysis for the heptane pool fire system.

Table 6.1: Data from all five simulations for the consistency analysis. The results for the steady, global mass flux ($kg/m^2 - sec$) and the average incident radiation flux (W/m^2) at the pool center, and at pool radii of 2, 4, 6, 8, 10, 12, 13, and 14 centimeters are shown.

	Base-Case	**Case-1**	**Case-2**	**Case-3**	**Case-4**
Soot Parameter	1.00	0.10	0.10	10.0	10.0
Mass Transfer Coefficient Parameter	1.00	0.85	1.15	0.85	1.15
Global Mass Flux	0.03264	0.01850	0.01845	0.04452	0.04524
Radiation (center)	25,518	15,726	16,201	35,855	36,130
Radiation (2 cm)	25,510	15,726	15,981	35,833	36,167
Radiation (4 cm)	25,470	15,794	15,935	35,730	36,228
Radiation (6 cm)	25,464	16,090	16,036	35,611	36,093
Radiation (8 cm)	25,138	16,080	16,049	35,295	35,832
Radiation (10 cm)	24,500	15,158	15,108	34,839	35,405
Radiation (12 cm)	22,789	13,807	13,777	33,493	34,008
Radiation (13 cm)	22,086	13,824	13,237	32,498	33,014
Radiation (14 cm)	21,268	12,686	12,607	31,429	31,918

Table 6.2: Consistency results for tests on the whole 10-member dataset. Each test assumes different levels of experimental error. The percentages along the top of the table represent mass burn rate error, and those along the left side of the table represent radiative heat flux error.

	10%	12%	15%	17%	20%
20%	X	X	X	X	X
25%	X	X	X	X	X
30%	X	X	X	X	X
35%	X	X	X	X	√
40%	X	X	√	√	√

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

7.1 Summary

Pool fires are an important example of combustion systems that are frequently studied in order to prevent costly damage and preserve human life. The study of these and other types of combustion systems can greatly benefit from the application of computational science. If properly developed, computational models can reveal new information on these systems and prove useful in predicting fire behavior. The information gathered can lead to better safety measures and policies in dealing with and preventing accidental fires. Additionally, computational models can potentially do all this without the level of expense involved with experimentation.

In order to be useful, computational models must be developed with an understanding of their limitations. The calculations from computational models are not without error, and this error must be understood and quantified as far as possible. In the case of liquid pool fire simulations, the Responsive Boundary model is developed in order to address the issue of boundary condition error with the fuel inlet boundary. The model seeks to predict the behavior of the liquid pool as it vaporizes fuel to feed the flame and receives thermal input from the flame. In its development, the Responsive Boundary model seeks to capture enough of the important physics involved to be able to give accurate predictions of the mass flux of fuel to the fire and still be simple enough as to not add unnecessary computational burden to the fire code with which it works.

The Responsive Boundary model consists of simple mass, energy, and in the case of multi-component fuels, component balances that are expressed as continuum equations. The model tracks the temperature profile in the liquid fuel as a function of depth. It also accounts for the loss of mass as the liquid pool evaporates. Mass transfer correlations predict the the mass fluxes, which are strongly dependent on the temperature of the liquid

at the pool surface. The model also includes the ability to make predictions for fuels consisting of more than one liquid species as transportation fuels are often mixtures of hundreds to thousands of chemical species.

In addition to the development of the model, effort is made to verify and validate its calculations. Verification activities such as the use of manufactured solutions and grid convergence tests are applied to the boundary model to establish a degree of confidence that the code functions without any problems arising from programming errors and to ensure that the error arising from discretization methods for the continuum models are what they should be. The tests show that the model can reproduce manufactured solutions and that the order of error with respect to spatial discretization is the expected value. The time step discretization error is of lower order than expected. This is thought to be the result of additional error arising from the interpolation steps that allow calculation of the temperature profile as the liquid domain shrinks. Despite this loss of accuracy the order of error is still sufficient to show that the discretized solution to the model approaches the continuum solution as the step sizes become arbitrarily small. In short, the applied verification activities demonstrate that the computer algorithm implemented in the boundary model produces the correct solution to the continuum relations that are the foundation of the model within an acceptable level of error.

Sensitivity analysis of the boundary model's inputs suggest that factors like wind speed and thermal input to the pool have the greatest influence on the desired output of the model: the mass flux from the pool surface. Once incorporated with the fire code (ARCHES) a series of grid convergence tests demonstrate that error arising from spatial discretization in the gas domain is small compared to the modeling error when simulations apply 200^3 nodes to the domain for 90^3 cubic centimeter computational simulation of heptane and methanol pool fires similar those that Klassen and Gore (1992) work with in their experiments.

The primary tool of validation used to evaluate modeling error is the technique of data collaboration described by Feeley et al. (2004). This technique compares the range of model output given the uncertainty in model input to experimental data with their uncertainty. A mathematical definition of consistency is applied to indicate what qualifies as agreement

between model and experiment. The importance of data collaboration is found in how both the computational and experimental facets of research come together to better arrive at an understanding of the physical phenomena involved. An example of data collaboration is provided in ARCHES simulations of small buoyant helium plumes and experimental data taken from the work of Schönbucher et al. (1986). This collaboration seeks to find agreement between model predictions and experimental measurements of fringe order in holographic images of helium plumes. Although initial inconsistencies are found between the experimental and computational data, further sensitivity analyses lead to revisions in the evaluation of the experimental data that leads to a consistent dataset.

The method of data collaboration is finally applied to pool fire simulations done using ARCHES with the Responsive Boundary model giving inlet boundary predictions. The simulations are based on and compared with data from small heptane fires done by Klassen and Gore (1992). The quantity of comparison is initially chosen to be the mass flux from the pool surface at steady burning conditions. The radiative heat fluxes to the pool surface at steady conditions are eventually included in the analysis to more fully investigate model/experimental agreement. The model parameters thought to contain the greatest degree of uncertainty and to have the greatest influence on the outputs are the soot parameter from ARCHES and the mass transfer coefficients from the boundary model. A series of simulations are run to test the range of possible model output, and they are compared with the experimental data. The initial consistency test shows strong consistency with the mass burn rate, and there is also good consistency with each of the radiation fluxes when examined individually. However, there is inconsistency for the whole dataset. The inconsistency arises between the mass flux and certain radiation fluxes. The Responsive Boundary is generating a fuel flux that is in agreement with experiment, but it is doing so using incident radiation flux predictions from ARCHES that are too high when compared with experiment. The inconsistency is especially pronounced between the mass flux and the radiative fluxes near the outer edges of the pool. The most likely reason for the inconsistency is thought to be a systematic bias error in the radiometers used in the original experiments. The radiation values seem to be under-measured. The conclusion from the consistency analysis is that the boundary model does give output that

is consistent with experimental data given the likely sources of error in the experimental radiation measurements.

The results from the heptane and methanol pool fire comparisons are encouraging. Despite the differences in the thermophysical properties between the two fuels, the mass flux predictions for both cases are comparable to the experimental measurements. In addition, many of the characteristics observed in pool fires are also well reproduced in the simulations with the Responsive Boundary. These trends include the dominance of radiation heat transfer in determining the steady mass flux, the secondary role of convective heat transfer and its correlation with flame shape during the different phase of the fire, the role of volatility and the fuel vapor dome over the pool surface in influencing the burn rate at the different phases of the fire, and the behavior of the liquid temperature during the fire.

7.2 Conclusions

The overall conclusions associated with this present study are: (1) The Responsive Boundary model addresses the need for a realistic fuel inlet boundary condition for use with liquid pool fire simulations. The boundary model captures the necessary physics of the pool system through application of the pool energy and mass balances. The model also applies mass transfer correlations for buoyant and wind-driven systems for prediction of the evaporative mass flux from the pool surface. (2) Verfication exercises such as manufactured solutions and grid convergence analyses show that the computer algorithm implementation of the boundary model produces mass flux data that are consistent with the proper solutions (within acceptable error) to the continuum equations that form the basis of the boundary model, and the code's results for the tested calculations are not deterred by any obvious programming errors. (3) Model validation through consistency analysis has shown that despite initial discrepancies, consistency is found between experimental measurement and computational predictions from ARCHES of interference fringe order from holographic images of small, laminar helium plume systems. Through this analysis a better understanding of the error involved with analysis of holographic images is achieved. (4) Consistency analysis also demonstrates agreement between experimental data and

predictions from the ARCHES/boundary model for the medium-sized heptane pool fire system when likely levels of experimental error arising from mass flux measurement and radiative heat flux measurement are considered.

7.3 Future Work

This project represents a considerable amount of effort to develop and test the reliability of the Responsive Boundary model, but it is only a beginning. The present work is limited in the type and size of pool fires tested thus far, and this suggests at least three directions that must be pursued in further development of the boundary model: (1) The model must be tested with larger pool fires, (2) The model must be tested using different fuels including multi-component fuels and surrogates, and (3) the model must also must be tested for systems of significant cross-wind.

This study applies data from Klassen and Gore (1992) as a starting point for verification and validation tests with the boundary model because the system of interest that they provide is sufficiently small to allow more rigorous simulations with small discretization error without overloading available computational resources. The pool fires tested are large enough that the important modes of heat transfer are dominant. In other words, although the 30-centimeter fires simulated are small compared to most real life accidental liquid fuel fires, they are large enough to capture some important characteristics of the larger fires that are the intended use of pool fire simulation tools. Nevertheless, a logical subsequent step is to simulate larger pool fires in order to gage the effectiveness of the Responsive Boundary model for use in such scenarios. The JP-8 pool fire data that Blanchat et al. (2004) provide would be useful in such a series of tests. The primary issues with larger fire simulations is the limitation of computational resource and time. For the simulations in this project a resolution of 200^3 nodes is found to be sufficient. To simulate a 1-meter fire while maintaining the same spatial discretization would require more than 650^3 nodes, and performing such a large simulation requires almost 35 times the number of processors (the 200^3-node simulations use 512 processors.) Running a 1-meter pool fire simulation with present resources would likely require running with a coarser resolution, and this implies a larger discretization error in the results. With the continued advancement and

improvement of computational technology, the possibility of simulating larger fires with more desirable resolution in the future is likely, but it must wait.

Another possibility of future work for the Responsive Boundary model is its application with other types of fuels including fuel mixtures. Most fuels thought likely to be involved in large transportation fire scenarios are complex mixtures. The use of surrogate models for these types of fuels in simulations is promising, but like all approximations the error inflicted with surrogates must be evaluated. The Responsive Boundary model has means to work with multi-component fuels as described in Chapter 3, but this ability is not yet thoroughly tested. Preference is given in this report to simpler, single-component pool fire systems because it is better to start validating a model with its simpler applications first. Another important reason that the multi-component capabilities of the boundary model are not yet tested is found in the use of reaction/mixing tables in fire codes like ARCHES. These tables provide tabular information on important physical properties in the pool fire system as functions of certain computed properties like mixture fraction and heat loss. The use of these tables allow predictions that normally come from complex kinetic reaction mechanisms and turbulent mixing models without the limitations of stiffness in differential equation solution methods. The tables are processed prior to the fire simulations, and as presently constituted the simulations use only one set of tables per simulation. One of the parameters used in the formation of the table is the fuel inlet composition. This means that the use of tabulation requires the assumption of constant inlet composition. For single-component fuel simulation this poses no problems, but multi-component fuels change composition as they evaporate. For simulations involving the boundary model some way must be found to allow for changing composition in the fire code, or error must be allowed in the assumption that the change in composition will not significantly affect the thermodynamic properties of the flame.

The last scenario for which the boundary model is intended that requires further attention is the scenario of a transportation fire under windy conditions. As most accidental fires occur outdoors, there is a greater chance of windy conditions being present than for the absence of such conditions. Buoyant conditions are tested first because of the greater simplicity in their modeling. Winds deflect the pool fire's flames significantly, and the

simulation of this type of fire requires a larger domain to capture the whole fire, which means more computational resource is needed. The presence of wind blowing over the pool introduces a different mode of convection (forced) that requires use of a different mass transfer correlation. The use of a different correlation merits further validation work for these types of pool fire simulations.

REFERENCES

[1] AIAA. Guide for the Verification and Validation of Computational Fluid Dynamics Simulations. *Tech. Rep. AIAA-G-077-1998*, AIAA, Reston, Virginia. 1998.

[2] Akita K, Yumoto T. Heat Transfer in Small Pools and Rates of Burning of Liquid Methanol. In: *Tenth Symposium (International) on Combustion*. 1965; p. 943.

[3] Alger RS, Corlett RC, Gordon AS, Williams FA. Some Aspects of Structures of Turbulent Pool Fires. *Fire Technology*. 1979;15:142–156.

[4] Associated Press. "Highway Interchange Collapse Threatens to Paralyze San Francisco Traffic Grid." *FoxNews.com.* 30 April 2007. Web. 10 October 2009. ⟨http://www.foxnews.com/story/0,2933,269156,00.html⟩.

[5] Bird RB, Stewart WE, Lightfoot EN. *Transport Phenomena*. New York, NY: John Wiley and Sons, 2nd ed. 2002.

[6] Blanchat T, Humphries L, Gill W. Sandia Heat Flux Gauge Thermal Respone and Uncertainty Models. *Tech. Rep. SAND2000-1111*, Sandia National Laboratories. 2000.

[7] Blanchat TK, Nicolette VF, Sundberg WD, Figueroa VG. Well-Characterized Open Pool Experiment Data and Analysis for Model Validation and Development. *Tech. Rep. SAND2006-7508*, Sandia National Laboratories. 2006.

[8] Blinov V, Khudyakov G. Diffusion Burning of Liquids. *Tech. rep.*, Izdatel'stvo Akademii Nauk SSSR, Moscow (Moscow, Academy of Sciences). 1957.

[9] Blinov, V., and Khudyakov, G. Diffusion Burning of Liquids. *Izdatel'stvo Akademii Nauk*. SSSR, Moscow. English Translation: U.S. Army Engineering Research and Development Laboratories, Information Resources Branch, Translation Analysis Section, Fort Belvoir, VI, No. T-1490a-c. 1961.

[10] Brown AL, Gill W, Lopez C. Predictive Evolution of Fuel from a Liquid Pool Fire: Phenomenology Identification and Ranking Exercize. In: *ASME International Mechanical Engineering Congress and Exposition*. 2006.

[11] Brown AL, Vembe BE. Evaluation of a Model for the Evaporation of Fuel from a Liquid Pool in a CFD Fire Code. In: *ASME International Mechanical Engineering Congress and Exposition*. IMECE. 2006.

[12] Burgess D, Strasser A, Grumer J. Diffusive Burning of Liquid Fuels in Open Trays. *Fire Research Abstracts and Reviews*. 1961;3:177–192.

[13] Centauri L. Uncertainty Quantification for Holographic Interfereographic Images. Masters thesis, University of Utah. 2010.

[14] Chatris J, Planas E, Arnoldos J, Casal J. Effects of Thin-Layer Boilover on Hydrocarbon Pool Fires. *Combustion Science and Technology*. 2001;171:141–161.

[15] Ciddor PE. Refractive Index of Air: New Equations for the Visible and Near Infrared. *Applied Optics*. 1996;35:1566–1573.

[16] Constantinides A, Mostoufi N. *Numerical Methods for Chemical Engineers with MATLAB Applications*. Upper Saddle River, NJ: Prentice Hall. 1999.

[17] Corlett R, Fu T. Some Recent Experiments with Pool Fires. *Pyrodynamics*. 1966; 1:253–269.

[18] Dittus F, Boelter L. Heat transfer in automobile radiators of the tubular type. *University of California Publications in Engineering*. 1930;2:443.

[19] Drysdale D. *An Introduction to Fire Dynamics*. Chichester, UK: John Wiley and Sons, 2nd ed. 1998.

[20] Eddings EG, Yan S, Ciro W, Sarofim AF. Formulation of a Surrogate for the simulation of Jet Fuel Pool Fires. *Combustion Science and Technology*. 2005;177:715–739.

[21] Feeley R, Seiler P, Packard A, Frenklach M. Consistency of a Reaction Dataset. *Journal of Physical Chemistry*. 2004;108:9573–9583.

[22] Ferziger J, Peric M. *Computational Methods for Fluid Dynamics*. Berlin, Germany: Springer-Verlag, 3rd ed. 2002.

[23] Fletcher D, Kent J, Apte V, Green A. Numerical Simulations of Smoke Movement from a Pool Fire in a Ventilated Tunnel. *Fire Safety Journal*. 1994;23:305–325.

[24] Gardiner W, Hidaka Y, Tanzawa T. Refractivity of Combustion Gases. *Combustion and Flame*. 1981;40:213–219.

[25] Gothard F, Clobanu M, Breban D, Bucur C, Sorescu G. Predicting the Parameters in the Wilson Equations for Activity Coefficients in Binary Hydrocarbon Systems. *Industrial and Engineering Chemistry, Process Design and Development*. 1976;15:333–337.

[26] Hamins A, Fischer S, Kashiwagi T, Klassen M, Gore J. Heat Feedback to the Fuel Surface in Pool Fires. *Combustion Science and Technology*. 1994;97:37–62.

[27] Hatton L. The T Experiments: Errors in Scientific Software. *IEEE Computational Science and Engineering*. 1997;4(2):27–38.

[28] Hottel HC. Review of Certain Laws Governing Diffusive Burning of Liquids. *Fire Research Abstracts and Reviews*. 1958;1:41–44.

[29] Incropera FP, DeWitt DP. *Fundamentals of Heat and Mass Transfer*. New York, NY: John Wiley and Sons, 5th ed. 2002.

[30] Joulain P. The Behavior of Pool Fires: State of the Art and New Insights. In: *Twenty-Seventh Symposium (International) on Combustion/ The Combustion Institute*. 1998; pp. 2691–2706.

[31] Joulain P. Convective and Radiative Transport in Pool and Wall Fires: 20 Years of Research in Poitiers. *Fire Safety Journal*. 1996;26:99–149.

[32] Klassen M, Gore J. Structure and Radiation Properties of Pool Fires. *Tech. Rep. NIST-GCR-94-651*, National Institutes of Standards and Technology. 1992.

[33] Klassen M, Gore J, Sivathanu Y, Hamins A, Kashiwagi T. Radiative Heat Feedback in a Toluene Pool Fire. In: *Twenty-Fourth Symposium (International) on Combustion/ The Combustion Institute*. 1992; pp. 1713–1719.

[34] Koseki H. Large Scale Pool Fires: Results of Recent Experiments. In: *Fire Safety Science-Proceedings of the Sixth International Symposium*. 1999; pp. 115–132.

[35] Mansfield CR, Peck ER. Dispersion of Helium. *Journal of the Optical Society of America*. 1969;59:199–204.

[36] Nakos J, Kleltner N. The Radiative-Convective Partitioning of Heat Transfer to Structures in Large Pool Fires. In: *1989 National Heat Transfer Conference*, vol. HTD-Vol. 106. 1989; pp. 381–387.

[37] Nakos J. Uncertainty Analysis of Steady State Incident Heat Flux Measurements in Hydrocarbon Fuel Fires. *Tech. Rep. SAND2005-7144*, Sandia. 2005.

[38] Nakos, J.T. Personal communication, January 17, 2011.

[39] Novozhilov V, Koseki H. CFD Prediction of Pool Fire Burning Rates and Flame Feedback. *Combustion Science and Technology*. 2004;176:1283–1307.

[40] O'Hern T, Weckman E, Gerhart A, Tieszen S, Schefer R. Expeirmental Study of a Turbulent Buoyant Helium Plume. *Journal of Fluid Mechanics*. 2005;544:143–171.

[41] Oberkampf W, Trucano T. Verification and Validation in Computational Fluid Dynamics. *Progress in Aerospace Sciences*. 2002;38:209–272.

[42] Patankar S, Spaulding D. A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows. *International Journal of Heat and Mass Transfer*. 1972;15:1787–1806.

[43] Poling BE, Prausnitz JM, O'Connell JP. *The Properties of Gases and Liquids*. New York, NY: McGraw-Hill, 5th ed. 2001.

[44] Prasad K, Li C, Kailasanath K, Ndubizu C, Ananth R, Tatem PA. Numerical modeling of methanol liquid pool fires. *Combustion Theory and Modeling*. 1999; 3:743–768.

[45] Randsalu EJ, Lam CS, Weckman EJ, Brown AL, Gill W. Measurement of Fuel Regression Rates in a Jet Fuel Pool Fire in Crosswind. In: *Proceedings of HTFED04, 2004 ASME Heat Transfer/Fluids Engineering Summer Conference*. 2004.

[46] Rasbash D, Rogowski Z, Stark G. Properties of Fires of Liquids. *Fuel*. 1956;35:94–107.

[47] Reichenberg D. New Simplified Methods for the Estimation of the Viscosities of Gas Mixtures at Moderate Pressures. *Tech. rep.*, National Engineering Laboratory Report in Chemistry. 1977.

[48] Richardson L. The Deferred Approach to the Limit. *Transactions of the Royal Society of London, Series A*. 1927;226:299–361.

[49] Richardson L. The Approximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, with an Application to the Stresses in a Masonry Dam. *Transactions of the Royal Society of London, Series A*. 1910; 210:307–357.

[50] Roache PJ. *Verification and Validation in Computational Science and Engineering*. Albuquerque, NM: Hermosa Publishers. 1998.

[51] Schönbucher A, Arnold B, Banhardt V, Bieller V, Kasper H, Kaufmann M, Lucas R, Schiess N. Simultaneous Observation of Organized Density Structures and the Visible Field in Pool Fires. In: *Twenty-First Symposium (International) on Combustion/ The Combustion Institute*. 1986; pp. 83–92.

[52] Seader J, Henley E. *Separation Process Principles*. New York, NY: John Wiley and Sons Inc. 1998.

[53] Shinotake A, Koda S, Akita K. An Experimental Study of Radiative Properties of Pool Fires of an Intermediate Scale. *Combustion Science and Technology*. 1985;43:85–97.

[54] Sibulkin M. Estimates of the Effect of Flame Size on Radiation from Fires. *Combustion Science and Technology*. 1973;7:141–143.

[55] Spinti J, Thornock J, Eddings E, Smith P, Sarofim A. *Transport Phenomena in Fires*, chap. Heat Transfer to Objects in Pool Fires, pp. 69 – 136. Boston, MA: WITpress. 2006.

[56] Takeno K, Hirano T. Behavior of Combustible Liquid Soaked in Porous Beds Durning Flame Spread. In: *Twenty-Second Symposium (International) on Combustion/ The Combustion Institute*. 1988; pp. 1223 – 1230.

[57] Takeno K, Hirano T. Flame Spread over Porous Solids Soaked with a Combustible Liquid. In: *Twenty-first Symposium (International) on Combustion/ The Combustion Institute*. 1986; pp. 75–81.

[58] Taylor R, Krishna R. *Multicomponent Mass Transfer*. Wiley Series in Chemical Engineering. New York, NY: John Wiley and Sons, 1st ed. 1993.

[59] Thomas P, Baldwin R, Heselden A. Buoyant Diffusion Flames: Some Measurements of Air Entrainment, Heat Transfer, and Flame Merging. In: *Tenth Symposium (International) on Combustion/ The Combustion Institute*. 1965; pp. 983–996.

[60] Thomson G, Brobst K, Hankinson R. An improved Correlation for Densities of Compressed Liquids and Liquid Mixtures. *AICHE Journal*. 1982;28:671–676.

[61] Turns SR. *An Introduction to Combustion*. Boston, MA: McGraw-Hill, 2nd ed. 2000.

[62] United States Fire Administration. "QuickStats, The Overall Fire Picture - 2008." U.S. Fire Administration Home Page, 27 August 2009. Web. 10 October 2009. ⟨http://www.usfa.dhs.gov/statistics/quickstats/index.shtm⟩.

[63] Violi A, Yan S, Eddings E, Sarofim A, Granata S, Faravelli T, Ranzi E. Experimental Formulation and Kinetic Model for JP-8 Surrogate Mixtures. *Combustion Science and Technology*. 2002;174:399–417.

[64] Walas SM. *Phase Equilibria in Chemical Engineering*. Boston, MA: Butterworth-Heinemann. 1985.

[65] Wilke C, Lee C. Estimation of Diffusion Coefficients for Gases and Vapors. *Industrial and Engineering Chemistry*. 1955;47:1253.